CONFIDENCE-CONDITIONED SLOT COMPOSITION FOR STRUCTURED RETRIEVAL UNDER NOISY GROUNDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reliability-weighted fusion has been widely used, from products of experts (Hinton, 2002) and gated multimodal units (Arevalo et al., 2017) to trusted multi-view classification (Han et al., 2021). Yet, how per-slot confidence should enter a structured retrieval query, and whether it should be used during training, is rarely examined. In this work, we systematically study CCSC (Confidence-Conditioned Slot Composition) for intent-to-knowledge-graph retrieval, comparing eight alternative composers over 37 matched runs on five benchmarks built from BINS, CoDEx-M, WN18RR, and FB15k-237. Synthetic confidence that perfectly separates correct from wrong slots isolates how confidence is used from how well it is estimated. Our central finding is that a simple design, CCSC, which adds slot confidence as a log-bias to query-side attention logits, consistently improves retrieval: it achieves higher mean reciprocal rank (MRR) than role-wise matching, pooling, SumMLP, and learned scalar weighting on all five benchmarks (by 0.014–0.027 over the strongest), and than a model trained without confidence on all five when both receive the same confidence. Ablations trace the gain to two factors: (1) training with confidence in the attention logits, since a confidence-only variant already recovers most of the gain, and (2) learned content scores, which help further when correct and wrong slots are mixed. Notably, we find that a model trained with confidence should also receive it at deployment, as removing it reverses its advantage over the confidence-free model on four benchmarks. We release code and checkpoints to facilitate future research.

1 INTRODUCTION

A growing body of work on intent-based network management translates high-level requests into policies, network actions, or structured intent representations (Dzeparoska et al., 2023; Manias et al., 2024; Li et al., 2025b). We study a modular intent-to-knowledge-graph (KG) retrieval setting with two stages. An upstream *grounding pipeline* identifies semantic roles and maps them to entities or relations with per-slot confidence. A downstream *retriever* composes those grounded inputs into a query against an ontology-bounded candidate set. This paper studies the second stage: given role-labeled, confidence-annotated slots, how should retrieval combine useful information while attenuating unreliable components?

Contrastive dual encoders provide a natural backbone (Karpukhin et al., 2020; Radford et al., 2021; Wang et al., 2022). However, a role-agnostic sum treats different positions symmetrically, and a composer that ignores confidence has to infer reliability from the embeddings alone. Ordered concatenation preserves positions, but still needs a rule for their contribution. The design question is how to combine role correspondence, supplied reliability, and learned content into a useful retrieval representation.

We propose CCSC, a contrastive retrieval framework that composes noisy, confidence-annotated grounded slots into queries for bounded structured retrieval. Its four processing steps (Figure 1) retain the modular structure of the application: (a) receive grounded slots and confidence; (b) compute confidence-conditioned query-side role weights; (c) bind roles to matching candidate positions; and (d) use a shared projection to compare the resulting query and candidate representations. The method operates on provided embeddings and confidence, not raw text. Even a signal that identifies reliable slots leaves a composition choice: confidence-only weighting, learned content weighting,

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.

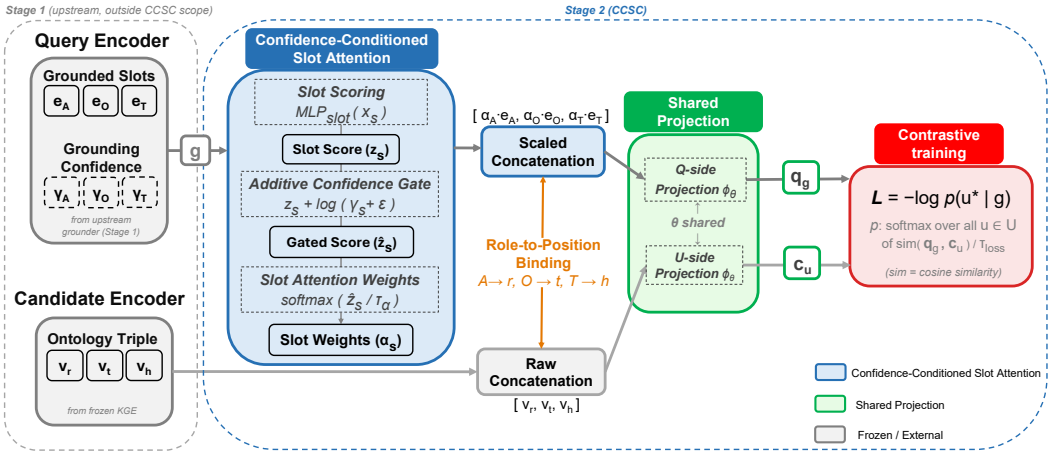

Figure 1: CCSC's two-stage architecture. The upstream grounder and frozen KG embeddings supply the inputs. Within Stage 2, confidence-conditioned weights scale the query roles, and ordered concatenation keeps role positions; the query and unweighted candidate share a projection. The loss box shows the retrieval term of Eq. 3. The contrastive objective trains the downstream modules, not the frozen KG embeddings.

and their combination can produce different query representations. We evaluate the complete CCSC composer against the specified alternatives, then compare procedures under the same supplied confidence to assess the added value of learned weighting. Fixed-checkpoint interventions examine dependence on evaluation confidence and responses to its transformation.

Our paper makes three contributions, each connecting a design choice to its measured consequences:

(i) Confidence-conditioned role composition. We combine content scores and supplied reliability before scaled, role-preserving concatenation. Matched comparisons on five controlled panels show higher mean MRR than the evaluated role-wise matching, pooling, SumMLP, and learned-scalar alternatives, all of which omit supplied confidence. The confidence-aware GateOnly comparison measures the smaller added value of learned weighting. Role alignment also supports a shared projection with fewer learned-module parameters; we report its accuracy tradeoffs.

(ii) Confidence use and fitted weighting. Under the same confidence-on evaluation, CCSC has higher mean MRR than NoConf on all five panels; removing evaluation confidence reverses their mean ordering on four. Thus the comparison of these training procedures depends on how the signal is used at retrieval time. GateOnly comparisons separately locate the smaller, condition-dependent gain from the procedure with learned content weights.

(iii) Condition-sensitive findings. Preserving the order of supplied confidence does not preserve retrieval performance: squaring confidence at evaluation reduces MRR for a fixed CCSC even while synthetic Brier/ECE improve. Incorrect and missing slots also yield different benefit patterns, and confidence gains remain at temperature 1.0. Figures 2 and 3 show why the weighting rule and the input condition must be evaluated together, beyond confidence discrimination alone.

Our controlled study separates the value of receiving informative confidence from the value of composing with it. The resulting comparisons give concrete evidence for confidence-use and weighting choices within known-catalog retrieval; realistic grounder confidence remains a separate evaluation question.

2 Method

2.1 Problem Formulation and Role Binding

We study Stage 2 of a two-stage intent-to-triple system. A pre-grounded query is $g = ((e_A, \gamma_A), (e_O, \gamma_O), (e_T, \gamma_T))$, where $s \in \{A, O, T\}$ indexes action, object, and target roles, e_s

is a frozen-lookup embedding, and $\gamma_s \in [0, 1]$ is supplied confidence. A candidate $u = (h, r, t)$ contains a head entity, relation, and tail entity. Retrieval selects an entire candidate from a fixed set $\mathcal{U} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ over entity and relation vocabularies $\mathcal{E}$ and $\mathcal{R}$; it is not a masked-entity completion task.

The role binding is $A \to r$, $O \to t$, and $T \to h$. Query roles (A, O, T) and candidate positions (r, t, h) therefore occupy matching blocks of the projection input. Frozen TransH knowledge graph embeddings (KGE) (Wang et al., 2014) supply 256-dimensional vectors on both sides. We learn the downstream scoring and projection modules. Let x_s and v_r, v_t, v_h denote the individually ℓ_2-normalized role vectors. Normalization and missing-input conventions, together with the provenance of the TransH choice, are recorded in Appendix B.

2.2 Confidence-conditioned query weighting

Let $\mathcal{P} \subseteq \{A, O, T\}$ denote the roles present in g. The following three steps weight the query inputs; they correspond to the scoring, confidence gate, and slot-weight blocks in Figure 1.

Slot scoring. A shared slot MLP maps each normalized role vector to an unnormalized content score, not a calibrated correctness probability:

$$z_s = \mathrm{MLP}_{\mathrm{slot}}(x_s). \tag{1a}$$

The slot MLP does not inspect other slots; the subsequent shared projection processes the weighted roles jointly.

Confidence conditioning. We add the supplied confidence as a log bias, with $\varepsilon = 10^{-6}$:

$$\hat{z}_s = z_s + \log(\gamma_s + \varepsilon). \tag{1b}$$

Role weighting. Whenever a query has at least one present role ($\mathcal{P} \neq \varnothing$), we convert the confidence-conditioned logits $\{\hat{z}_s\}_{s\in\mathcal{P}}$ into nonnegative role weights that sum to one over its present roles:

$$\alpha_s = \frac{\exp(\hat{z}_s/\tau_\alpha)}{\sum_{j\in\mathcal{P}} \exp(\hat{z}_j/\tau_\alpha)}, \qquad s \in \mathcal{P}. \tag{1c}$$

Absent roles have weight zero whenever $\mathcal{P} \neq \varnothing$. When $\mathcal{P} = \varnothing$, CCSC assigns $1/3$ to each role and retains the query with its fixed missing-role vectors. Based on prior development results, we fixed the central role temperature at $\tau_\alpha = 0.5$; other temperatures are examined separately.

Scaled concatenation and a shared projection. The weights scale query slots before ordered concatenation in query order (A, O, T). Candidate vectors are concatenated in the matching order (r, t, h) and receive no attention weighting:

$$\begin{aligned} q_g &= \mathrm{norm}(\phi_\theta([\alpha_A x_A; \alpha_O x_O; \alpha_T x_T])), \\ c_u &= \mathrm{norm}(\phi_\theta([v_r; v_t; v_h])). \end{aligned} \tag{2}$$

Both sides use the same ϕ_θ; norm denotes output ℓ_2 normalization with the numerical conventions in Appendix B. Ordered blocks retain role positions before projection; whole-model performance gaps do not isolate this property alone.

Appendix B.1 explains the equivalent normalized product of content and confidence factors over the same role index.

2.3 Training and comparisons

Retrieval score and loss. The score compares the final query and candidate vectors: $s(g, u) \equiv \mathrm{sim}(q_g, c_u) = q_g^\top c_u$. For target $u^* \in \mathcal{U}$, the retrieval loss compares it with the complete candidate bank, including u^* in the denominator:

$$\mathcal{L}_{\mathrm{retr}}(g, u^*) = -\log \frac{\exp(s(g, u^*)/\tau_{\mathrm{loss}})}{\sum_{u'\in\mathcal{U}} \exp(s(g, u')/\tau_{\mathrm{loss}})}. \tag{3}$$

Sharpening term. With $H(\alpha) = -\sum_{s\in\{A,O,T\}} \alpha_s \log \alpha_s$ and $0 \log 0 = 0$, we minimize

$$\mathcal{L}(g, u^*) = \mathcal{L}_{\text{retr}}(g, u^*) + \lambda_\alpha H(\alpha). \tag{4}$$

Training averages this objective over each minibatch. Here $\tau_{\text{loss}} = 0.1$ differs from τ_α, and $\lambda_\alpha = 0.01$. The positive entropy term encourages sharpening when minimized. Appendix B retains the numerical log floor.

All methods receive the same frozen inputs and candidates. **Role-wise CM** (cosine matching) averages aligned role cosines; **Pooled CM** takes cosine after averaging raw role vectors. Both are training-free and omit confidence. **SumMLP** uses a role sum and shared projection; **LSW** (learned scalar weighting) learns global role logits at temperature 0.1. **Uniform** learns a projector over ordered, equally weighted present roles. **GateOnly** sets $z_s = 0$ but learns the projection. **NoConf** trains without the log-confidence term; its original evaluation also omits it, while Table 3 additionally tests supplying confidence to that fitted model. **Separate** unties the initially shared query and candidate projections. Appendix B gives the detailed definitions, entropy paths, and all-absent conventions.

3 Experiments

3.1 Benchmarks and evaluation protocol

BINS-derived application panels. The Business Intent and Network Slicing (BINS) dataset supplies our networking application (Li et al., 2025a;b). The curated/narrow tiers contain 175/53 candidate triples and fixed, disjoint intent-row partitions: 21,351/11,974 training rows and 2,669/1,497 final rows; the equally sized reserved validation partitions are unused here. One annotated triple is sampled and fixed per intent and repeat. Its relation, tail, and head supply the action, object, and target slots before synthetic corruption; no text grounder is evaluated. For example, $(h, r, t) =$ (user type, allow service, network service) yields $A =$ allow service, $O =$ network service, and $T =$ user type. These are derived retrieval splits, not the original named-entity-recognition split. Five repeats share one frozen TransH per tier while varying initialization, sampled gold, corruption, and order. They are not five independently trained backbones.

Larger graph-derived catalogs. CoDEx-M (Safavi & Koutra, 2020), WN18RR (Dettmers et al., 2018), and FB15k-237 (Toutanova & Chen, 2015) supply three different TRAIN triple collections. Each derived catalog contains 8,192 known triples, all seen by the frozen backbone and by Stage-2 training. A fixed set of 1,024 target triples with two observations each supplies learning-rate selection; a disjoint set of 2,048 targets with four observations each supplies the final 8,192-row panel. These are new corrupted observations of known candidate facts, not native KG-completion test results. Three backbone seeds each contain three Stage-2 optimization repeats. FB15k-237 was adopted after its results were observed; subsequent controls and stress tests on existing panels are post-hoc comparisons.

Corruption and confidence. The central condition draws correct, wrong, or missing states with probabilities $0.60/0.25/0.15$ per role. Wrong slots use typed alternatives from the frozen embedding vocabulary; missing slots use the fixed replacement and presence mask. Correct slots have confidence 1 and wrong slots have cosine-derived confidence in $[0.3, 0.6]$; missing slots are masked. The cosine compares the gold and replacement embeddings in the same frozen space used to select wrong neighbors. Thus the signal uses gold information in its construction and perfectly separates correctness among present slots, without revealing the correct identities for wrong or missing roles. This controlled signal is not measured grounder confidence. Appendix A gives the exact recipes.

Training, selection, and metrics. All Stage-2 fits run 30 epochs with batch size 64, Adam, fixed loss temperature 0.1, and terminal-epoch evaluation. BINS uses learning rate 0.001 with no early stopping or Stage-2 hyperparameter search. Each learned graph method selects from $\{0.0003, 0.001, 0.003\}$ using two tuning seeds on one backbone, then keeps that choice for final evaluation. Uniform/SumMLP use the complete selection budget after a numerical correction, documented in Appendix B.

MRR averages reciprocal ranks; Hits@K measures the fraction ranked within K. BINS uses optimistic exact ties, and graph panels use average exact ties. Overall graph summaries first average the four views per target, then Stage-2 repeats within a backbone, then the three backbone means. Their reported SD is across those three means. BINS SD is across its five paired repeats. Neither SD nor the number of query rows is a confidence interval for cross-dataset generalization. Mean orderings are descriptive, not claims of statistical significance on every panel. Complete MRR and Hits@1/3/10, paired values, and 13 corruption/presence groups are supplied with the result tables.

3.2 Value against alternative composition methods

Table 1 situates CCSC against role-wise cosine matching, pooling, SumMLP, and learned scalar weighting on the same input panels. CCSC has higher mean MRR than these specified alternatives on all five panels. These whole-procedure comparisons favor CCSC over the specified alternatives, which do not use supplied confidence; Table 2 then asks what CCSC adds over a learned method receiving that same signal.

Table 1: Mean MRR of CCSC and four alternatives that do not use supplied confidence; all use the same observations and candidate order (definitions: Section 2.3; GateOnly: Table 2). Training-free CM has one value per backbone or BINS repeat. Bold: highest mean in each row. SD and Hits: Appendix C.

Dataset/tier	Role-wise CM	Pooled CM	SumMLP	LSW	CCSC
BINS curated	0.6597	0.5990	0.7163	0.7418	**0.7684**
BINS narrow	0.6221	0.6183	0.7235	0.7352	**0.7580**
CoDEx-M	0.6043	0.4224	0.6879	0.7163	**0.7303**
WN18RR	0.6526	0.6526	0.8083	0.8261	**0.8460**
FB15k-237	0.6311	0.4240	0.6968	0.7449	**0.7622**

Role-wise CM already compares aligned roles without training; Pooled CM instead averages raw role vectors before cosine. Each uses the same fixed observations and candidate order as its corresponding CCSC comparison; graph CM supplies one reference per backbone, not three independent Stage-2 fits. Appendix B gives both definitions.

3.3 Confidence, content, and sharing

Table 2 compares separately trained weighting procedures. Under their original evaluation rules, confidence-informed GateOnly is closer to CCSC in mean MRR than NoConf or Uniform, which omit supplied confidence. The CCSC$-$GateOnly mean difference ranges from -0.0002 to $+0.0052$: narrow BINS favors GateOnly on average, and WN18RR retains two negative nested-repeat differences.

Table 2: Ablation study at the central setting: CCSC minus each separately fitted variant in MRR, with paired sample SD beneath each mean difference. GateOnly uses confidence weights; NoConf omits confidence in training and in this evaluation; Uniform uses equal present-role weights; Separate unties the projections. Positive differences favor CCSC. Absolute MRR, its SD, and Hits appear in Appendix C; sharing intervals are in Appendix E.

Dataset/tier	GateOnly	NoConf	Uniform	Separate
BINS curated	$+0.0038$ ± 0.0015	$+0.0278$ ± 0.0028	$+0.0271$ ± 0.0034	$+0.0010$ ± 0.0013
BINS narrow	-0.0002 ± 0.0017	$+0.0249$ ± 0.0027	$+0.0245$ ± 0.0021	-0.0034 ± 0.0037
CoDEx-M	$+0.0052$ ± 0.0013	$+0.0126$ ± 0.0009	$+0.0146$ ± 0.0014	$+0.0081$ ± 0.0005
WN18RR	$+0.0014$ ± 0.0014	$+0.0209$ ± 0.0008	$+0.0192$ ± 0.0014	$+0.0011$ ± 0.0008
FB15k-237	$+0.0024$ ± 0.0013	$+0.0129$ ± 0.0007	$+0.0162$ ± 0.0011	$+0.0047$ ± 0.0009

Content value across input conditions. For inputs containing both correct and wrong roles, mean CCSC−GateOnly MRR differences are $+0.0064$ on BINS curated, $+0.00001$ on BINS narrow, $+0.0088$ on CoDEx-M, $+0.0022$ on WN18RR, and $+0.0028$ on FB15k-237. The narrow-tier mean remains near zero. At the central setting, both CCSC and GateOnly attain MRR 1 on all-correct inputs in every panel; the zero gap reflects this metric ceiling. With zero correct roles, mean MRR instead favors GateOnly on BINS narrow and FB15k-237, with CCSC−GateOnly differences of -0.0027 and -0.0009, respectively. These results locate added value and its exceptions within the role-input conditions, rather than attributing the whole Table 1 gain to learned content scores. The groups overlap and this is a post-hoc description, not a rule for selecting future tasks. Appendix C.1 gives the same groups on every panel, including missingness and saved SD.

Confidence use at training and evaluation. Table 3 compares the two training procedures under each common evaluation condition. Adding confidence only at evaluation improves NoConf's mean MRR on every panel, bringing it closer to CCSC. With evaluation confidence enabled, CCSC has higher mean MRR than NoConf supplied with the same confidence on each panel. Omitting evaluation confidence reverses the mean ordering on four panels; CoDEx-M is the exception. Training with confidence and adding it only at evaluation therefore produce different mean outcomes even under the same evaluation signal.

Table 3: MRR under common evaluation signals. Each Δ is CCSC minus NoConf, with paired sample SD beneath the difference. ON supplies confidence to both fitted models; OFF omits the log-confidence term while retaining each checkpoint's scorer and projector. Bold: higher mean in each pair. Absolute-cell SD and Hits appear in Table 8.

	Evaluation confidence ON			Evaluation confidence OFF		
Panel Training signal	NoConf without	CCSC with	Δ_{ON}	NoConf without	CCSC with	Δ_{OFF}
BINS curated	0.7632	**0.7684**	$+0.0053$ ± 0.0017	**0.7407**	0.7103	-0.0303 ± 0.0054
BINS narrow	0.7522	**0.7580**	$+0.0058$ ± 0.0014	**0.7331**	0.6976	-0.0356 ± 0.0043
CoDEx-M	0.7233	**0.7303**	$+0.0070$ ± 0.0004	0.7177	**0.7207**	$+0.0030$ ± 0.0003
WN18RR	0.8446	**0.8460**	$+0.0014$ ± 0.0010	**0.8252**	0.8239	-0.0013 ± 0.0010
FB15k-237	0.7592	**0.7622**	$+0.0030$ ± 0.0010	**0.7493**	0.7463	-0.0030 ± 0.0009

Within each checkpoint, this intervention retains content scores, projection, mask, and candidate bank and omits or includes the log-confidence term. Across training procedures, learned parameters, entropy gradients, and selected learning rates can differ, so this is a comparison of fitted procedures rather than a pure training-only effect. On curated BINS, removing evaluation confidence lowers mean MRR by 0.0581 for CCSC and 0.0225 for NoConf. The reversal thus also reflects CCSC's greater dependence on evaluation confidence. Appendix D separates this contrast from the four separately trained content/confidence variants.

Usefulness of fitted weights. Replacing only the weights of frozen NoConf models with uniform weights tests their usefulness directly; results and temperature-dependent reversals are reported in Appendix D.1.

Sharing: size and accuracy. The Separate column of Table 2 addresses a distinct design choice. The shared and Separate implementations contain 329,728 and 592,896 learned-module parameters, a 44.39% reduction. Shared has higher mean MRR on the three graph catalogs and BINS curated; narrow BINS favors Separate by 0.0034. This is a measured size–accuracy tradeoff, with no adopted noninferiority margin. CoDEx-M selects different learning rates, so that panel compares selected procedures rather than tying alone. Appendix E gives the uncertainty assumptions and the supplied results retain all 37 matched pairs and 20 post-hoc intervals.

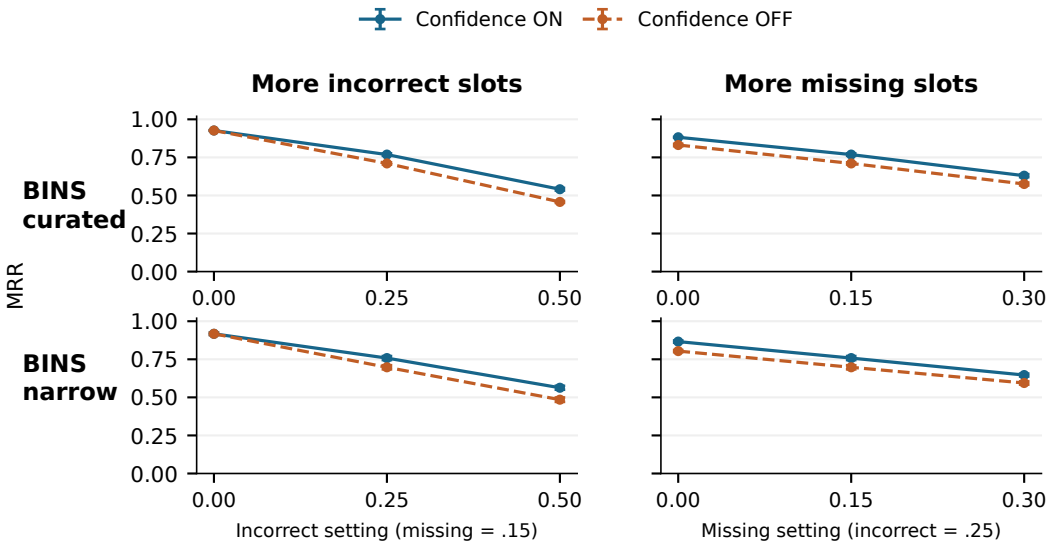

Figure 2: Incorrect and missing slots produce different confidence-benefit patterns in BINS. Columns vary these two settings separately for the same fitted CCSC. Confidence ON uses the original supplied values; confidence OFF here sets present-role confidence to one. Checkpoint and observations are fixed within each comparison; observations vary across conditions. Bars show the stored SD over five BINS repeats. The default point is reused between sweeps. Full five-panel results and paired differences appear in Appendix A.1.

3.4 Corruption, temperature, and confidence quality

Figure 2 examines how the benefit of supplied confidence changes with incorrect and missing slots. In the tested wrong-slot sweep the ON/OFF gain increases even as absolute MRR falls. Missingness has a different pattern: the curated gain is nonmonotone and the narrow gain decreases across the tested settings. The graph panels show the same increasing confidence benefit with the wrong-slot setting, with smaller gains than BINS at the nonzero wrong-slot settings (Appendix A.1). These contrasting sweeps separate the value of supplied confidence under incorrect inputs from its value when information is absent.

Temperature provides a separate test. At matched training and evaluation temperature 1.0, CCSC$-$NoConf mean MRR remains positive on all five panels (Table 12). Benefits therefore do not require the confidence exponent ten induced by $\tau_\alpha = 0.1$. The extra benefit over GateOnly remains small or negative on BINS, however. Low temperature also produces reversals: on CoDEx-M at 0.1, CCSC$-$GateOnly is -0.0869. A ten-point evaluation-only curve over 0.1–1.0 and the separately trained $0.1/0.5/1.0$ anchors answer different questions (Appendix F.2); neither proves that 0.5 is optimal.

Quality interventions retain the same observations and checkpoint while changing supplied confidence. Monotone powers, clipping, and within-role permutation distinguish confidence magnitude from its association with correctness. Full present-role permutation makes CCSC worse than confidence OFF on each panel: the mean MRR differences range from -0.0184 on CoDEx-M to -0.0416 on BINS curated, with the other panels also negative. Squaring preserves confidence ordering but lowers mean MRR relative to the original supplied confidence on all five panels; on CoDEx-M it also falls below confidence OFF.

Figure 3 shows a concrete mismatch between slot-level quality and retrieval. Under this synthetic construction, squaring lowers wrong-slot Brier error from γ^2 to γ^4 and lowers ECE, while correct slots remain at one. Perfect separation is preserved, so AUROC also remains one. These slot-quality changes are expected; the retrieval decreases in Figure 3 are empirical outcomes. Table 13 connects identity, square, and confidence-OFF retrieval with slot-quality metrics on the same inputs at one

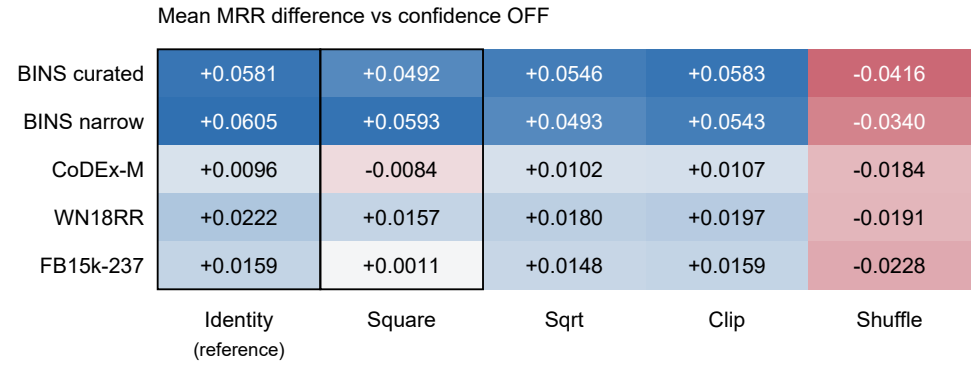

Figure 3: Evaluation-time confidence transforms on all five benchmarks: paired mean MRR minus that of the same CCSC with confidence OFF (positive = better than OFF; "(reference)" = untransformed confidence). Outlined: identity vs. square. Per-run slot metrics: Table 13; SD: Figure 7.

illustrative coordinate per panel. Squaring also changes the scale used by the composer's weighting rule. Square roots also lower mean MRR relative to identity on four panels, with CoDEx-M the exception. Both are evaluation-time changes to the confidence distribution of a fixed retriever. These results show sensitivity to signal use, not a causal effect of calibration or a comparison of models retrained with each transform. Better synthetic slot-level metrics alone therefore do not establish better retrieval for the existing checkpoint. The full map and Appendix F.3 give the tested scope beyond those illustrative coordinates.

Attention during learning. Learning curves and fixed-input validation traces in Appendices F.1 and F.4 describe how weights form. At the central temperature, NoConf can be less concentrated than CCSC despite lower training MRR. Concentration alone therefore does not diagnose poor retrieval; the frozen-NoConf interventions instead test the utility of its fitted weights directly.

Together, the comparisons support CCSC against the specified alternatives (Table 1), while locating its smaller, condition-dependent differences from GateOnly under the same confidence (Table 2). They also motivate reporting evaluation-confidence conditions with training-procedure comparisons (Table 3) and checking confidence transformations against retrieval outcomes as well as slot-quality metrics (Figure 3).

4 Related Work

Contrastive and structured retrieval. Dense passage retrieval and CLIP learn aligned dual encoders (Karpukhin et al., 2020; Radford et al., 2021). SimKGC uses contrastive bi-encoders for KG completion (Wang et al., 2022). Mehmood et al. (2024) use KGE-based service prediction and intent validation to map networking intents to services. GQE, Query2Box, and BetaE embed logical queries for answering over incomplete KGs (Hamilton et al., 2018; Ren et al., 2020; Ren & Leskovec, 2020). CCSC instead retrieves a whole known triple from three provided roles. KG question-answering pipelines also use entity-linking scores in downstream ranking (Yih et al., 2015); here the supplied scores enter the composition of a retrieval vector. CCSC combines supplied reliability and learned content while preserving query–candidate role correspondence.

Learned fusion and supplied reliability. Gated Multimodal Units learn input-dependent multiplicative gates for feature fusion (Arevalo et al., 2017). Trusted Multi-View Classification combines view-specific evidential predictions using Dempster–Shafer fusion (Han et al., 2021). Quality-aware Multimodal Fusion uses uncertainty to weight unimodal predictions and analyzes conditional generalization bounds (Zhang et al., 2023). In composed image retrieval, WISER uses verifier confidence for candidate/path fusion and refinement (Wang et al., 2026). CCSC instead combines a supplied role-confidence input with learned content scores before ordered concatenation and retrieval projec-

tion. NoConf and GateOnly distinguish confidence use from the additional value of learned content weighting.

Slot attention and gating. Slot Attention discovers exchangeable object-centric slots through iterative competition (Locatello et al., 2020); CCSC weights fixed, externally assigned roles in one pass. QASA derives slot quality from attention, selects slots during training, and uses log-gating in its Transformer decoder; its stated inference protocol omits the training selection mask (Ouyang et al., 2026). Qiu et al. (2025) study learned feature gates on scaled-dot-product-attention outputs. These works provide gating antecedents. CCSC's log bias also admits an equivalent normalized multiplicative form under the conditions specified in Appendix B.1; the empirical question here is how supplied role reliability and learned content are used in structured retrieval.

Confidence and downstream utility. Decision calibration connects calibration guarantees to specified loss functions and decision rules (Zhao et al., 2021). Our contribution is a concrete test of supplied role confidence in structured retrieval: the same-evaluation comparison of two training procedures changes order on four panels when confidence is removed, and a rank-preserving transformation can lower a fixed composer's MRR despite better slot-level error metrics. These measured effects specify how confidence use matters in this composer, rather than proposing the general connection between calibration and downstream utility as new.

5 Conclusion

We presented CCSC, a contrastive composition framework combining supplied role confidence, ordered query–candidate alignment, and a shared projection. Controlled BINS and graph-derived retrieval comparisons show efficacy against the specified composition alternatives, while distinguishing that result from the smaller, condition-dependent gain over confidence-only weighting. Direct training–evaluation comparisons show higher CCSC mean MRR than NoConf under common confidence-on evaluation on five panels, with the ordering reversed on four when that signal is removed. The sharing study measures a parameter reduction and its accuracy tradeoff; temperature and quality interventions reveal both surviving benefits and reversals. Together, these results support confidence-conditioned composition for fixed-role retrieval with informative confidence, while clarifying how its utility depends on signal use in the tested settings. Whether these benefits persist with naturally predicted roles and confidence remains untested.

6 Limitations and future work

Our experiments keep three roles, frozen TransH embeddings, and known candidate facts. Synthetic confidence strongly encodes corruption status. It does not establish native-grounder calibration, systematic or correlated errors, variable arity, or unseen-fact prediction. The baseline set contains a confidence-only learned retriever, but does not establish superiority over confidence-weighted matching or exact-match filtering. We do not isolate role-preserving concatenation, confidence-injection form, or entropy regularization in matched ablations. We have not evaluated a controlled sweep of partially informative, overlapping correct/wrong confidence distributions; the permutation stress test does not characterize that intermediate regime. The graph uncertainty rests on three backbone clusters, and BINS repeats are conditional on one backbone per tier. Additional analyses reuse exposed panels; FB15k-237 was adopted after observing its results. Sharing intervals are post-hoc and do not establish noninferiority; parameter counts do not establish lower latency or memory. Testing real grounder outputs and stronger cross-domain or schema claims requires separate evidence.

AI USE STATEMENT

In this work, we used generative AI tools for implementing methods, including debugging, finding related work, numerical audits, and experimental orchestration; generating the synthetic evaluation data, by helping implement the slot-corruption and confidence generators and the scripts that sample and deduplicate the graph-derived candidate catalogs from existing benchmark triples; providing feedback on experiments during planning and execution; and interpreting retained results. We have not used generative AI tools for designing the research methodology or experiments, developing the theoretical model or conceptual framework, proposing or refining hypotheses, formulating the mathematical claims in Appendices B.1 and F.3 or deriving and writing their proofs, cleaning or reformatting the BINS data or constructing its tiers, or translating the manuscript, and qualitative and thematic data analysis is not applicable to this work. Additionally, we used generative AI tools for drafting and editing the manuscript, preparing tables and figures, and preparing the anonymous code artifact. We have reviewed all AI-assisted work. The authors specified the experimental design, including the corruption and confidence recipes, baselines, ablations, and evaluation protocol, and approved every change to it; each AI suggestion was reviewed and adopted, modified, or rejected by the authors. The corruption probabilities, confidence ranges, and catalog-selection rules were set by the authors, and generated observations were checked against these specifications. Reported metrics come from executed evaluation procedures and retained artifacts, not numerical answers generated by language models; checks of AI-assisted work included source and artifact identity checks and bounded reproduction checks. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

The study evaluates retrieval from frozen structured candidate sets and does not execute network policies or other real-world actions. Its controlled corruption protocol should not be interpreted as a validated safety or reliability model for deployed grounding systems. Releasing data and code requires preserving the respective dataset and software terms while excluding private operational and review material.

REPRODUCIBILITY STATEMENT

Appendix A specifies the data units, corruption, selection, and ranking rules; Appendix B records the exact method and numerical conventions. The accompanying anonymous code artifact documents its included inputs, model states, commands, and verification scope. Two corrected training runs were selected before their new results and repeated in a fresh environment on the same host; the compared computation states and outputs matched exactly. Appendix G gives the check's scope and the unused NumPy state exception. This bounded re-execution does not establish independent reproduction of the full study.

REFERENCES

John Arevalo, Thamar Solorio, Manuel Montes-y-Gómez, and Fabio A. González. Gated multimodal units for information fusion. In *International Conference on Learning Representations (Workshop Track)*, 2017. URL https://arxiv.org/abs/1702.01992v1.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2D knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pp. 1811–1818, 2018. doi: 10.1609/aaai.v32i1.11573. URL https://ojs.aaai.org/index.php/AAAI/article/view/11573.

Kristina Dzeparoska, Jieyu Lin, Ali Tizghadam, and Alberto Leon-Garcia. LLM-based policy generation for intent-based management of applications. In *2023 19th International Conference on Network and Service Management (CNSM)*, pp. 1–7, 2023. doi: 10.23919/CNSM59352.2023.10327837. URL https://arxiv.org/abs/2402.10067v1. AnServApp 2023.

William L. Hamilton, Payal Bajaj, Marinka Zitnik, Dan Jurafsky, and Jure Leskovec. Embedding logical queries on knowledge graphs. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/hash/ef50c335cca9f340bde656363ebd02fd-Abstract.html.

Zongbo Han, Changqing Zhang, Huazhu Fu, and Joey Tianyi Zhou. Trusted multi-view classification. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2102.02051.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. doi: 10.1162/089976602760128018. URL https://doi.org/10.1162/089976602760128018.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Jie Li, Sai Zou, Hongfeng Gao, Yanglong Sun, and Wei Ni. Business intent and network slicing correlation dataset, 2025a. URL https://doi.org/10.6084/m9.figshare.27311538.v1. Dataset, version 1; repository license: CC0 1.0.

Jie Li, Sai Zou, Yanglong Sun, Hongfeng Gao, and Wei Ni. Business intent and network slicing correlation dataset from data-driven perspective. *Scientific Data*, 12(1):419, 2025b. doi: 10.1038/s41597-025-04736-z. URL https://www.nature.com/articles/s41597-025-04736-z.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI)*, pp. 2181–2187, 2015. doi: 10.1609/aaai.v29i1.9491. URL https://ojs.aaai.org/index.php/AAAI/article/view/9491.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/8511df98c02ab60aea1b2356c013bc0f-Abstract.html.

Dimitrios Michael Manias, Ali Chouman, and Abdallah Shami. Towards intent-based network management: Large language models for intent extraction in 5G core networks. *arXiv preprint arXiv:2403.02238*, 2024. URL https://arxiv.org/abs/2403.02238v2. Version 2, 22 May 2024; presented at DRCN 2024.

Kashif Mehmood, Katina Kralevska, and David Palma. Knowledge graph embedding in intent-based networking. In *2024 IEEE 10th International Conference on Network Softwarization (NetSoft)*, pp. 13–18, 2024. URL https://arxiv.org/abs/2405.07850v1. WIN 2024 workshop; author preprint arXiv:2405.07850v1.

Tianran Ouyang, Xingping Dong, Jing Zhang, Mang Ye, Jun Chen, and Bo Du. QASA: Quality-guided K-adaptive slot attention for unsupervised object-centric learning. *arXiv preprint arXiv:2601.12936*, 2026. URL https://arxiv.org/abs/2601.12936v1. Version 1, 19 January 2026.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Gated attention

for large language models: Non-linearity, sparsity, and attention-sink-free. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3345. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/904e89bb4e632e75fb47f093b620b257-Abstract-Conference.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Hongyu Ren and Jure Leskovec. Beta embeddings for multi-hop logical reasoning in knowledge graphs. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://snap.stanford.edu/betae/.

Hongyu Ren, Weihua Hu, and Jure Leskovec. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *International Conference on Learning Representations*, 2020. URL https://snap.stanford.edu/query2box/.

Tara Safavi and Danai Koutra. CoDEx: A comprehensive knowledge graph completion benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8328–8350. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.669. URL https://aclanthology.org/2020.emnlp-main.669/.

Zhiqing Sun, Shikhar Vashishth, Soumya Sanyal, Partha Talukdar, and Yiming Yang. A re-evaluation of knowledge graph completion methods. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5516–5522, 2020. doi: 10.18653/v1/2020.acl-main.489. URL https://aclanthology.org/2020.acl-main.489/.

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

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. SimKGC: Simple contrastive knowledge graph completion with pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4281–4294. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.acl-long.295. URL https://aclanthology.org/2022.acl-long.295/.

Tianyue Wang, Leigang Qu, Tianyu Yang, Xiangzhao Hao, Yifan Xu, Haiyun Guo, and Jinqiao Wang. WISER: Wider search, deeper thinking, and adaptive fusion for training-free zero-shot composed image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16865–16875, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Wang_WISER_Wider_Search_Deeper_Thinking_and_Adaptive_Fusion_for_Training-Free_CVPR_2026_paper.html.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28, pp. 1112–1119, 2014. doi: 10.1609/aaai.v28i1.8870. URL https://ojs.aaai.org/index.php/AAAI/article/view/8870.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 1321–1331, 2015. doi: 10.3115/v1/P15-1128. URL https://aclanthology.org/P15-1128/.

Qingyang Zhang, Haitao Wu, Changqing Zhang, Qinghua Hu, Huazhu Fu, Joey Tianyi Zhou, and Xi Peng. Provable dynamic fusion for low-quality multimodal data. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 41753–41769. PMLR, 2023. URL https://proceedings.mlr.press/v202/zhang23ar.html.

Shengjia Zhao, Michael P. Kim, Roshni Sahoo, Tengyu Ma, and Stefano Ermon. Calibrating predictions to decisions: A novel approach to multi-class calibration. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/hash/bbc92a647199b832ec90d7cf57074e9e-Abstract.html.

Reading guide: from the main results to supporting evidence.

- Figure 1 and the method: implementation and the role-weight product in Appendices B and B.1.
- Table 1: full MRR/Hits and the graph comparison in Appendix C, Tables 4–5.
- Table 2: absolute component results in Appendix C, input groups in Appendix C.1, and sharing intervals in Appendix E.
- Table 3: full confidence-grid metrics in Appendix D; frozen-NoConf replacement in Appendix D.1.
- Figures 2 and 3: full corruption results in Appendix A.1; confidence transformations in Appendix F.3.
- Temperature and attention analyses: Appendix F.2 for temperature, and Appendices F.1 and F.4 for training and fixed-development traces.

Appendix A specifies the data and ranking rules; Appendix G specifies reproduction and artifact scope.

A Protocol and reproducibility details

The supplementary reference tables keep datasets, methods, temperatures, replicates, and corruption conditions separate. Per-group denominators are the applicable query rows; groups may overlap. Empty groups are unavailable, not zero-valued performance. No dataset-averaged headline score is formed.

The 13 masks are all rows, all three roles correct, at least one correct and one wrong role, any missing role, zero correct roles, and the eight presence patterns. Presence bits follow relation/tail/head order. Within a corruption subgroup the metric is the mean over its selected rows, preserving the original row-mask reduction; it is not a new equal-target reweighting. Both row and target-anchor counts are retained.

The five evaluation corruption conditions in Figure 2 use $(p_{\text{wrong}}, p_{\text{missing}})$ equal to $(0, 0.15)$, $(0.25, 0)$, $(0.25, 0.15)$, $(0.25, 0.30)$, and $(0.50, 0.15)$; p_{correct} is the remaining probability. The third is the central condition. The first still contains missing roles, so it is not a fully uncorrupted input panel. Each condition has its own fixed observations for the paired confidence ON/OFF evaluation.

Graph-derived catalogs. The fixed known candidate set contains 8,192 deduplicated TRAIN triples per dataset. Candidate order and target partitions are fixed by the released recipes and identities. Stage-2 training observes every candidate fact once per epoch. Selection uses 1,024 target anchors with two views each; final evaluation uses a disjoint set of 2,048 anchors with four views each. Both partitions contain known TRAIN facts. The original CoDEx-M catalog also excludes a previously exposed development catalog by literal triple identity. This does not establish unseen entities or unseen KG-training facts.

The graph backbone seeds are 13, 71, and 167. Their respective Stage-2 seeds are (61001,61007,61027), (62003,62011,62017), and (63001,63029,63031). The learning-rate grid is $\{0.0003, 0.001, 0.003\}$, using tuning seeds 51001 and 51007 on backbone 13. Selection averages terminal validation MRR and retains the fixed tie preference 0.001, then 0.0003, then 0.003. The selected settings are not changed after final-panel results.

Correct graph slots have confidence one; absent slots have zero. For a wrong slot, confidence is $0.3 + 0.3\,\text{clip}((\cos(e, e') + 1)/2, 0, 1)$. Here e is the gold slot embedding and e' its replacement. The generator uses this gold-dependent geometry; it is not information obtained from a deployed grounder. Typed neighbors exclude the target and use stable tie ordering. The wrong/missing state and neighbor-rank draws are shared across backbone seeds, while each backbone supplies its own geometry. Thus backbone variation also changes the induced alternatives. These confidence values encode correctness by construction; they are not estimates learned from a separate calibration set.

BINS-derived tiers. The five repeat labels are 42, 0, 7, 123, and 2026. Each repeat binds its initialization, sampled gold, corruption, and order streams. One frozen TransH per tier is reused.

The candidate counts are 175 and 53, with 2,669 and 1,497 final intent rows. The package retains the backend's typed lookup and scalar arithmetic rather than substituting the graph implementation. Correct and missing confidences are one and zero; wrong confidence uses the same $[0.3, 0.6]$ cosine-derived range above, computed by the BINS scalar recipe and then stored in float32. The wrong neighbor is drawn from the top 30 typed alternatives with self exclusion and stable typed-ID tie order. Missing vectors are the fixed vocabulary means normalized with a 10^{-9} denominator offset. All-missing and all-wrong rows remain in the evaluation. Adam uses learning rate 0.001 and terminal epoch 30, without early stopping or validation selection.

Ranking. With $n_>$ strictly higher scores and $n_=$ scores equal to the target's, including the target itself, graph rank is $1 + n_> + (n_= - 1)/2$ and BINS rank is $1 + n_>$. Both rules use exact float32 ties. We report reciprocal rank and Hits@1/3/10 under each panel's declared rule. Tie handling can affect rank-based comparisons (Sun et al., 2020); the effect of harmonizing these two rules has not been measured here. Graph views are first averaged within target, then within backbone, then across backbones. SD across three backbone means is not the SD of nine independent datasets. The BINS mean and SD refer to its five paired repeat panels.

A.1 Complete corruption results

The main figure emphasizes the BINS application. Figure 4 retains all five panels and both absolute performance and paired confidence benefit. The declared graph settings are generator controls, not measured unconditional error frequencies. The central condition is shared between the wrong and missing sweeps, not an additional replicate.

With missingness fixed at 0.15, the paired mean ON–OFF gain increases as the wrong-slot setting rises from 0.25 to 0.50 on all five panels. At a wrong-slot setting of 0.25, increasing missingness from 0 to 0.30 reduces the endpoint gain on narrow BINS and all three graphs, but leaves curated BINS above its starting value. Both curated BINS and FB15k-237 show an intermediate rise followed by a decline. At nonzero wrong-slot settings, graph gains are smaller than those on both BINS tiers. With no wrong slots, setting present-role confidence to one leaves the supplied values unchanged, so ON and OFF coincide.

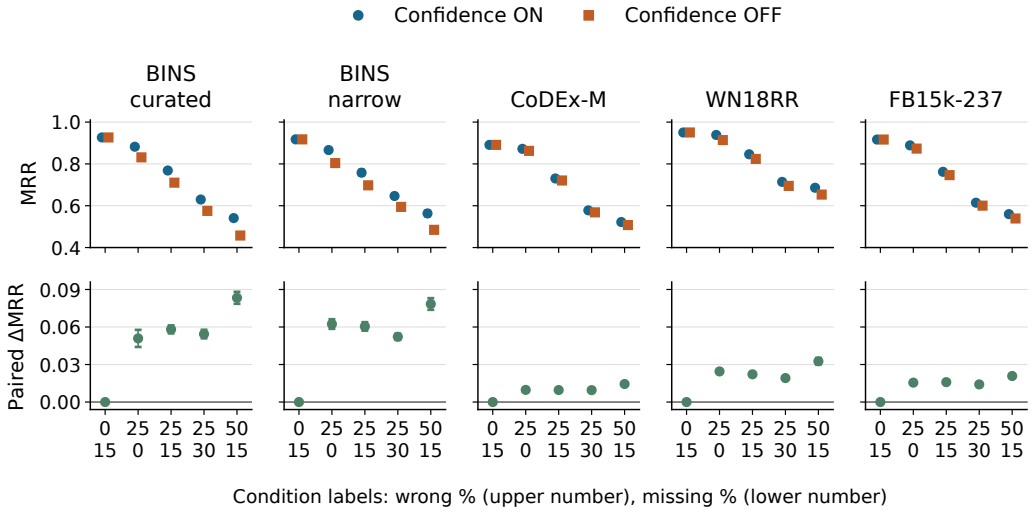

Figure 4: Full corruption results. Columns are the five panels; top shows confidence ON and OFF (present-role confidence set to one), and bottom shows their paired MRR difference. The categorical conditions list wrong/missing settings and are not one ordered calibration axis. Bars use saved sample SD under each panel's repeat hierarchy.

B IMPLEMENTATION, NUMERICAL VARIANTS, AND BASELINES

Backbone-selection provenance and scope. The TransH backbone family was retained following an earlier exploratory BINS comparison with TransE (Bordes et al., 2013) and TransR (Lin et al., 2015). This records a choice of model family under an earlier experimental configuration, not a selection experiment on the current checkpoint assets or evidence of optimality on the present catalogs. Each current composition comparison holds its frozen backbone fixed.

Graph module architecture. For dimension 256, the slot MLP (content scorer) is Linear(256,256), LayerNorm, Tanh, dropout 0.1, and a bias-free Linear(256,1). The shared projection is Linear(768,256), LayerNorm, Tanh, dropout 0.1, and Linear(256,256). Thus $\phi_\theta : \mathbb{R}^{768} \to \mathbb{R}^{256}$. The SumMLP projection instead uses widths 256–512–256. Compatible components use named deterministic initialization; linear weights use Xavier uniform and biases start at zero. These layer definitions and the retained BINS backend are included in the supplied code.

Role composition and baselines. The frozen TransH lookups supply unprojected entity vectors and relation translation vectors. Relation-plane normal vectors are retained for provenance but are not inputs to the composer; entities are not projected onto a relation-specific hyperplane here. The candidate branch uses the same projection as the query branch but does not run the query slot scorer. Role vectors are normalized before learned composition. Pooling CM instead averages the raw role vectors before a single normalization and cosine comparison, with its own 10^{-12} floor and fixed missing replacements. Role-wise CM averages three aligned cosines; it is a different baseline and is not relabeled as pooling CM. Both references use the same observed inputs and candidate order as their corresponding learned comparisons.

Weighting, sharing, and missing roles. GateOnly sets $z_s \equiv 0$; NoConf uses $\hat{z}_s = z_s$ in its original training and evaluation, with confidence insertion tested separately. Separate uses independently trained ϕ_{θ_q} and ϕ_{θ_c}; the other learned procedures share their query and candidate projection. Uniform sets $\alpha_s = 1/|\mathcal{P}|$ for $s \in \mathcal{P}$ when $\mathcal{P} \neq \varnothing$, with zero absent-role weights and zero weights when every role is absent. The softmax-based central models retain their uniform all-absent convention. GateOnly has no learned content score; any logged entropy term then has no gradient to the projector. NoConf and CCSC include the same positive entropy coefficient but need not have equal entropy gradients. LSW learns global logits, whereas SumMLP uses a sum and no learned attention entropy. These details are part of the fitted procedures, so we do not name their differences as pure one-factor causal effects.

Numerical settings. The entropy formula uses the continuous convention $0 \log 0 = 0$. The implementation evaluates the logarithm as $\log(\max(\alpha_s, 10^{-9}))$, multiplies by the unchanged α_s, and averages the per-query term over the minibatch. This numerical floor is separate from the confidence epsilon and vector-normalization rules.

Role-vector normalization uses $v/(\|v\|_2 + 10^{-8})$; thus "normalized" does not assert exact unit length for zero or tiny vectors. For central CCSC, output normalization in Eq. 2 applies $w = v/(\|v\|_2 + 10^{-8})$ and then $w/\max(\|w\|_2, 10^{-12})$, retaining zero vectors as zero. The corrected graph Uniform/SumMLP generation uses query outer-normalization epsilon 10^{-8} in training and evaluation, while retaining the inner normalization $x/(\|x\| + 10^{-8})$ and candidate outer epsilon 10^{-12}. Zero rows remain present, and no gradient clipping, dtype change, or epsilon search is introduced. This is not a claim of whole-function equivalence for arbitrarily small nonzero vectors.

The fixed-checkpoint confidence and weight interventions retain their separately recorded numerical path, specified in Appendix D.1.

Graph correction and reported results. A query-normalization correction addressed nonfinite optimizer state in some original graph Uniform/SumMLP runs. We repeated their entire learning-rate selection and confirmation budget, and report those corrected results. The affected-run and rerun counts are preserved in the accompanying numerical provenance notes. The other graph methods retain their original training: their evaluation either matches the saved scores with certified normalization equality or uses the prescribed corrected evaluation.

BINS extension and retained comparisons. The BINS Uniform/SumMLP/LSW extension adds 30 fits and 30 final evaluations, separately from the graph correction. Retained BINS controls preserve their query and score rows. Pooling comparisons are linked to the current CCSC results by matching inputs, candidate order, query rows, checkpoints, and scores.

Every new fit checks finite gradients, parameters, and Adam state and retains epochs 0–30. The accompanying numerical provenance notes and reference results distinguish original training, corrected training, and corrected evaluation for each displayed value.

B.1 Normalized product over role preferences

For a nonempty present-role set, rearranging Eq. 1c gives

$$\alpha_s \propto \exp(z_s/\tau_\alpha)(\gamma_s + \varepsilon)^{1/\tau_\alpha}, \qquad s \in \mathcal{P}. \tag{5}$$

At role temperature 1.0, the unnormalized confidence factor is $\gamma_s + \varepsilon$. At a fixed temperature, squaring replaces $(\gamma_s + \varepsilon)^{1/\tau_\alpha}$ by $(\gamma_s^2 + \varepsilon)^{1/\tau_\alpha}$ while retaining the content scores. Changing temperature rescales both factors. The content score can offset the confidence preference, so the highest-confidence role need not have the largest weight. Define two distributions over the same role index: $p_c(s) \propto \exp(z_s/\tau_\alpha)$ and $p_\gamma(s) \propto (\gamma_s + \varepsilon)^{1/\tau_\alpha}$, each normalized over $\mathcal{P}$. Then

$$\alpha_s = \frac{p_c(s)p_\gamma(s)}{\sum_{j\in\mathcal{P}} p_c(j)p_\gamma(j)}. \tag{6}$$

This is a product-of-experts form over roles (Hinton, 2002): the factors encode content preference and supplied-confidence weighting. It requires no conditional-independence assumption about correctness events. Neither factor is established as a calibrated role-correctness probability, and α is not a posterior over correct triples. With the same exponent, epsilon, present-role mask, and renormalization, post-softmax multiplication represents the same real-valued weights. The all-absent fallback is specified separately in the method.

C Complete performance and component results

Table 4 gives MRR and Hits for the main composition comparison; Table 5 summarizes seven methods' graph MRR. Table 6 gives absolute component results at the central setting, and Table 7 locates CCSC−GateOnly differences by input condition. Overlapping MRR entries, including repeated CCSC anchors, redisplay the same results rather than new runs.

Among the four methods in Table 4, CCSC has the highest mean on every reported metric and panel. Its mean advantage over LSW is larger for Hits@1 than for Hits@10 on each panel.

SD is the saved sample SD: five BINS repeats or three graph backbone means, with nested Stage-2 repeats retained in the machine-readable tables. All predefined corruption groups and favorable and unfavorable values remain available.

C.1 Additional content value by role-input condition

Table 7 expands the main text's central-temperature CCSC−GateOnly comparison using the same predefined state groups on all five panels. Both procedures receive the same observations and supplied confidence, but have separately learned parameters. Mixed inputs contain at least one correct and one wrong role; this does not measure semantic contradiction. Any-missing and zero-correct groups can overlap other groups, so they are not a partition whose means can be added.

The narrow mixed-input difference is small relative to its observed repeat variation. With zero correct roles, narrow BINS and FB15k-237 have mean MRR favoring GateOnly; other panels have positive means. Missingness likewise does not yield a uniform gain size. These descriptive differences locate where the CCSC procedure had an observed advantage in the tested panels, without establishing a rule predicting which new tasks favor CCSC. Exact paired values, nested repeats, and SD are copied in `conditioned_content_values_r1.json`, which also retains the saved row and target-anchor counts and metrics for all registered groups.

Table 4: MRR and Hits for Pooled CM, SumMLP, LSW, and CCSC. Full role-wise CM metrics are supplied in `rolewise_CM_full.json`. Sample SD appears beneath every mean and follows the panel-specific hierarchy and tie rule in Appendix A. Bold marks the highest displayed mean for each dataset/metric row, not significance. Exact values and per-condition denominators are included in the artifact.

Dataset/tier	Metric	Pooled CM	SumMLP	LSW	CCSC
BINS curated	MRR	0.5990 ±0.0094	0.7163 ±0.0075	0.7418 ±0.0074	**0.7684** ±0.0079
	Hits@1	0.4983 ±0.0087	0.6079 ±0.0085	0.6358 ±0.0091	**0.6673** ±0.0106
	Hits@3	0.6523 ±0.0131	0.7908 ±0.0084	0.8206 ±0.0100	**0.8504** ±0.0059
	Hits@10	0.7953 ±0.0118	0.9141 ±0.0048	0.9305 ±0.0044	**0.9389** ±0.0036
BINS narrow	MRR	0.6183 ±0.0135	0.7235 ±0.0119	0.7352 ±0.0128	**0.7580** ±0.0132
	Hits@1	0.5359 ±0.0177	0.6390 ±0.0162	0.6505 ±0.0139	**0.6838** ±0.0168
	Hits@3	0.6450 ±0.0136	0.7601 ±0.0127	0.7758 ±0.0142	**0.7912** ±0.0136
	Hits@10	0.7921 ±0.0074	0.9090 ±0.0072	0.9186 ±0.0080	**0.9205** ±0.0082
CoDEx-M	MRR	0.4224 ±0.0024	0.6879 ±0.0022	0.7163 ±0.0028	**0.7303** ±0.0022
	Hits@1	0.3731 ±0.0019	0.6327 ±0.0034	0.6617 ±0.0039	**0.6777** ±0.0029
	Hits@3	0.4355 ±0.0036	0.7186 ±0.0012	0.7458 ±0.0025	**0.7590** ±0.0017
	Hits@10	0.5144 ±0.0031	0.7840 ±0.0016	0.8148 ±0.0022	**0.8262** ±0.0017
WN18RR	MRR	0.6526 ±0.0049	0.8083 ±0.0024	0.8261 ±0.0015	**0.8460** ±0.0010
	Hits@1	0.5675 ±0.0031	0.7339 ±0.0038	0.7640 ±0.0025	**0.7942** ±0.0012
	Hits@3	0.7021 ±0.0075	0.8627 ±0.0019	0.8698 ±0.0005	**0.8815** ±0.0006
	Hits@10	0.7986 ±0.0059	0.9315 ±0.0015	0.9378 ±0.0012	**0.9399** ±0.0009
FB15k-237	MRR	0.4240 ±0.0031	0.6968 ±0.0010	0.7449 ±0.0016	**0.7622** ±0.0011
	Hits@1	0.3735 ±0.0037	0.6301 ±0.0019	0.6802 ±0.0028	**0.7021** ±0.0010
	Hits@3	0.4375 ±0.0024	0.7357 ±0.0023	0.7833 ±0.0016	**0.7998** ±0.0018
	Hits@10	0.5190 ±0.0026	0.8174 ±0.0040	0.8628 ±0.0028	**0.8684** ±0.0029

Table 5: Seven-method graph MRR comparison. Role-wise CM is distinct from Pooled CM; both appear in Table 1. Sample SD across three backbone means appears beneath each mean. Bold marks the highest displayed mean per dataset, not significance.

Dataset/tier	Role-wise CM	SumMLP	Uniform	LSW	GateOnly	NoConf	CCSC
CoDEx-M	0.6043 ±0.0028	0.6879 ±0.0022	0.7157 ±0.0032	0.7163 ±0.0028	0.7251 ±0.0009	0.7177 ±0.0030	**0.7303** ±0.0022
WN18RR	0.6526 ±0.0058	0.8083 ±0.0024	0.8269 ±0.0011	0.8261 ±0.0015	0.8446 ±0.0022	0.8252 ±0.0010	**0.8460** ±0.0010
FB15k-237	0.6311 ±0.0045	0.6968 ±0.0010	0.7460 ±0.0022	0.7449 ±0.0016	0.7598 ±0.0021	0.7493 ±0.0017	**0.7622** ±0.0011

Table 6: Component results at the central setting, grouped by dataset and metric. MRR includes saved sample SD beneath each mean; Hits are means. All methods are separately fitted. Bold marks the highest displayed mean in each dataset/metric row, including displayed ties, not significance. The BINS and graph repeat hierarchies follow Table 1.

Dataset/tier	Metric	CCSC	GateOnly	NoConf	Uniform	Separate
BINS curated	MRR	**0.7684** ±0.0079	0.7646 ±0.0083	0.7407 ±0.0087	0.7413 ±0.0080	0.7674 ±0.0084
	Hits@1	**0.6673**	0.6625	0.6338	0.6360	0.6653
	Hits@3	**0.8504**	0.8471	0.8207	0.8187	0.8496
	Hits@10	**0.9389**	0.9363	0.9296	0.9299	0.9388
BINS narrow	MRR	0.7580 ±0.0132	0.7582 ±0.0145	0.7331 ±0.0119	0.7335 ±0.0113	**0.7614** ±0.0135
	Hits@1	0.6838	0.6846	0.6474	0.6489	**0.6875**
	Hits@3	0.7912	0.7908	0.7741	0.7747	**0.7928**
	Hits@10	0.9205	0.9184	0.9180	0.9174	**0.9233**
CoDEx-M	MRR	**0.7303** ±0.0022	0.7251 ±0.0009	0.7177 ±0.0030	0.7157 ±0.0032	0.7222 ±0.0026
	Hits@1	**0.6777**	0.6735	0.6634	0.6624	0.6697
	Hits@3	**0.7590**	0.7527	0.7455	0.7430	0.7501
	Hits@10	**0.8262**	0.8191	0.8176	0.8127	0.8166
WN18RR	MRR	**0.8460** ±0.0010	0.8446 ±0.0022	0.8252 ±0.0010	0.8269 ±0.0011	0.8450 ±0.0017
	Hits@1	**0.7942**	0.7923	0.7624	0.7650	0.7920
	Hits@3	**0.8815**	0.8808	0.8701	0.8702	0.8813
	Hits@10	0.9399	0.9383	0.9375	0.9383	**0.9402**
FB15k-237	MRR	**0.7622** ±0.0011	0.7598 ±0.0021	0.7493 ±0.0017	0.7460 ±0.0022	0.7575 ±0.0018
	Hits@1	**0.7021**	0.6999	0.6858	0.6823	0.6966
	Hits@3	**0.7998**	0.7965	0.7880	0.7842	0.7946
	Hits@10	**0.8684**	0.8673	0.8637	0.8612	0.8669

Table 7: CCSC − GateOnly paired MRR by input condition at role temperature 0.5. Values are saved mean ± sample SD: three backbone means for graphs and five repeats for BINS. This is a post-hoc view of existing panels, not new confirmation. Positive values favor CCSC; zero all-correct differences describe these runs, not a general equivalence. Groups overlap.

Panel	All correct	Correct + wrong	Any missing	Zero correct
BINS curated	$+0.00000 \pm 0.00000$	$+0.00643 \pm 0.00253$	$+0.00504 \pm 0.00327$	$+0.00186 \pm 0.00495$
BINS narrow	$+0.00000 \pm 0.00000$	$+0.00001 \pm 0.00297$	$+0.00020 \pm 0.00296$	-0.00271 ± 0.02083
CoDEx-M	$+0.00000 \pm 0.00000$	$+0.00878 \pm 0.00308$	$+0.00401 \pm 0.00145$	$+0.00656 \pm 0.00226$
WN18RR	$+0.00000 \pm 0.00000$	$+0.00216 \pm 0.00111$	$+0.00037 \pm 0.00280$	$+0.00482 \pm 0.00449$
FB15k-237	$+0.00000 \pm 0.00000$	$+0.00279 \pm 0.00199$	$+0.00341 \pm 0.00247$	-0.00089 ± 0.00542

D CONFIDENCE USE AND FIXED-STATE INTERVENTIONS

The direct grid is complete on all five panels at role temperature 0.5. The four states are Y_{00} (NoConf/OFF), Y_{10} (CCSC/OFF), Y_{01} (NoConf/ON), and Y_{11} (CCSC/ON), grouped by evaluation condition, with ON first in the tables. OFF omits the full log-confidence term. Each checkpoint's content scorer, projection, observations, presence mask, and candidate bank remain fixed during its evaluation toggle. Selected learning rates and learned representations can differ between the two training procedures; on WN18RR their selected rates are 0.001 for NoConf and 0.0003 for CCSC.

Two implementations of confidence OFF. The two implementations of confidence OFF are equivalent in exact arithmetic for the same fitted composer, presence mask, all-absent fallback, and normalization rule. Setting present-role confidence to one adds a common $\log(1+\varepsilon)$ bias, which cancels under softmax; the direct grid instead omits the log-confidence term. Both retain the fitted scorer and projection. The stored CCSC MRR values match exactly at all 37 paired central coordinates, including each panel's mean and SD. This verifies those recorded MRR values, not bitwise equality of every score tensor. The numerical paths and their histories remain separately identified.

The confidence-grid contrast $(Y_{11} - Y_{10}) - (Y_{01} - Y_{00})$ is distinct from $(\mathrm{CCSC} - \mathrm{GateOnly}) - (\mathrm{NoConf} - \mathrm{Uniform})$ among four separately trained procedures. A positive value of the former does not cancel a negative value of the latter or establish a universal synergy. In particular, a large difference-in-differences can include large relative harm when confidence is removed.

Table 8: Complete direct confidence grid: MRR and Hits for the four training/evaluation combinations. Sample SD appears beneath each MRR mean; Hits are means. Bold marks the higher displayed mean within each ON or OFF evaluation pair, including displayed ties, not significance. These are the same repeats as Table 3; full SD, nested repeats, and groups remain in the reference JSON.

		Confidence ON		Confidence OFF	
Dataset/tier	Metric	NoConf	CCSC	NoConf	CCSC
BINS curated	MRR	0.7632 ±0.0086	**0.7684** ±0.0079	**0.7407** ±0.0087	0.7103 ±0.0069
	Hits@1	0.6617	**0.6673**	**0.6338**	0.5999
	Hits@3	0.8432	**0.8504**	**0.8207**	0.7888
	Hits@10	0.9353	**0.9389**	**0.9296**	0.9131
BINS narrow	MRR	0.7522 ±0.0131	**0.7580** ±0.0132	**0.7331** ±0.0119	0.6976 ±0.0135
	Hits@1	0.6758	**0.6838**	**0.6474**	0.6019
	Hits@3	0.7866	**0.7912**	**0.7741**	0.7426
	Hits@10	0.9186	**0.9205**	**0.9180**	0.8990
CoDEx-M	MRR	0.7233 ±0.0023	**0.7303** ±0.0022	0.7177 ±0.0030	**0.7207** ±0.0029
	Hits@1	0.6704	**0.6777**	0.6634	**0.6663**
	Hits@3	0.7508	**0.7590**	0.7455	**0.7507**
	Hits@10	0.8193	**0.8262**	0.8176	**0.8200**
WN18RR	MRR	0.8446 ±0.0018	**0.8460** ±0.0010	**0.8252** ±0.0010	0.8239 ±0.0011
	Hits@1	0.7920	**0.7942**	**0.7624**	0.7608
	Hits@3	0.8811	**0.8815**	**0.8701**	0.8685
	Hits@10	**0.9399**	**0.9399**	**0.9375**	0.9366
FB15k-237	MRR	0.7592 ±0.0021	**0.7622** ±0.0011	**0.7493** ±0.0017	0.7463 ±0.0021
	Hits@1	0.6987	**0.7021**	**0.6858**	0.6830
	Hits@3	0.7962	**0.7998**	**0.7880**	0.7842
	Hits@10	0.8682	**0.8684**	**0.8637**	0.8603

D.1 UNIFORM REPLACEMENT IN FROZEN NOCONF MODELS

This intervention changes only evaluation weights at each fitted NoConf state. We distribute mass equally over present roles; when none are present, we retain the original uniform fallback and

missing-role vectors. This differs from the separately trained Uniform method's zero-alpha all-absent rule. No role or only one present role therefore produces no weighting change under this particular intervention.

Table 9: Uniform replacement minus original-weight evaluation of the *same* NoConf model, paired MRR $\pm$ sample SD. Each column uses models trained at that role temperature; 0.5 is central. A negative difference favors the original learned weights.

Panel	$\tau_\alpha = 0.1$	$\tau_\alpha = 0.5$	$\tau_\alpha = 1.0$
BINS curated	$+0.1495 \pm 0.0671$	-0.0010 ± 0.0009	-0.0002 ± 0.0008
BINS narrow	$+0.0040 \pm 0.0075$	-0.0051 ± 0.0059	-0.0064 ± 0.0045
CoDEx-M	$+0.0807 \pm 0.0011$	-0.0021 ± 0.0002	-0.0022 ± 0.0011
WN18RR	$+0.0040 \pm 0.0020$	$+0.0006 \pm 0.0007$	$+0.0003 \pm 0.0013$
FB15k-237	-0.0023 ± 0.0002	-0.0031 ± 0.0003	-0.0035 ± 0.0010

In Table 9, original learned weights are favored on four central panel means; WN18RR has the opposite sign. This is a different four-panel set from the confidence-grid reversals, whose exception is CoDEx-M. The same-state result need not have the same direction as NoConf versus a separately trained Uniform model. At 0.1, uniform replacement improves mean MRR by 0.1495 on curated BINS and 0.0807 on CoDEx-M but reduces it on FB15k-237. On narrow BINS at 0.1, the MRR difference is $+0.004018$ while Hits@1 changes by -0.007482. These are not uniform improvements of every retrieval metric, nor an ablation of CCSC's content scorer.

Uniform replacement evaluates 37 NoConf checkpoints at each of three training temperatures, for 111 evaluations. Separately, the central confidence grid contains 37 paired coordinates times four cells: 38 newly evaluated cells and 110 reused cells. These studies add 149 evaluations and zero fits; the grid does not contain 148 independent models. These interventions use query outer-normalization epsilon 10^{-12}, with the model-specific inner normalization retained. This is separately identified from the corrected 10^{-8} generation above. The new queries are not assumed bit-identical across those numerical paths. Saved aggregates, nested repeats, and all registered groups are provided in `confidence_interventions.json` with the supplementary reference tables.

E SHARING AND UNCERTAINTY

Table 10 shows higher shared-model mean MRR on four of the five panels. Differences relative to Separate range from -0.0034 on narrow BINS to $+0.0081$ on CoDEx-M.

Separate copies the shared initial projection and subsequently trains two independent projection modules. Its initialization does not draw a new seed. The 27 graph and 10 BINS final pairs preserve the planned input and seed correspondence. Selection can still choose different learning rates; for example, CoDEx-M uses 0.001 for Separate and 0.003 for shared CCSC. Its comparison is therefore between selected procedures.

The display uses CCSC minus Separate throughout. The stored intervals use the opposite direction; means are negated and $[L, U]$ becomes $[-U, -L]$, with SD unchanged. This is a display conversion, not a new interval estimate. The 20 reported intervals cover four metrics on five panels and were computed post-hoc as $\bar{d} \pm t_{0.975,n-1} s_d/\sqrt{n}$, using paired-unit differences. Their nominal 95% coverage is not simultaneous coverage corrected for selection or multiple comparisons. Graph intervals depend on three backbone means, and BINS intervals on five repeat panels conditional on a single backbone. No loss margin was adopted before examining these results. We consequently report size and paired accuracy differences, including unfavorable values, without declaring equivalence or noninferiority.

Table 10: Shared CCSC minus Separate: matched MRR differences and post-hoc nominal 95% intervals. Nine graph pairs are nested in three backbone clusters; graph intervals use the three cluster means, not nine independent backbones. BINS uses five repeat panels. The margin for equivalence or noninferiority is unadopted, and the intervals are not simultaneous guarantees.

Panel	Pairs	CCSC − Separate	Nominal 95% interval
BINS curated	5	$+0.0010 \pm 0.0013$	$[-0.0007, +0.0027]$
BINS narrow	5	-0.0034 ± 0.0037	$[-0.0080, +0.0012]$
CoDEx-M	9	$+0.0081 \pm 0.0005$	$[+0.0069, +0.0093]$
WN18RR	9	$+0.0011 \pm 0.0008$	$[-0.0009, +0.0031]$
FB15k-237	9	$+0.0047 \pm 0.0009$	$[+0.0025, +0.0068]$

F Attention, temperature, and confidence interventions

F.1 Learning trajectories and measurement timing

Learning-time measurements use the same pre-update forward pass as the training loss. Observations and order change with epochs. Graph fixed-validation traces instead evaluate a fixed input panel after an epoch; the model state still changes. These timing and denominator differences are preserved in the supplied definitions and tables. One present role forces weight one and is distinguished from learned concentration among multiple present roles. Attention dispersion and maximum role mass are different statistics.

At later central-temperature epochs, NoConf can be less concentrated than CCSC on curated BINS and the graph catalogs despite lower training MRR. The early narrow-tier peak also subsides. These patterns describe fitted trajectories; they do not establish that confidence universally prevents harmful collapse or identify the entropy term as its cause.

Table 11 uses separately fitted models trained at role temperature 0.1 and fixed development inputs at epoch 30, distinct from the central-setting final evaluation in Section 3.3. On the displayed all-correct CoDEx-M inputs, CCSC has mean maximum role weight 0.875 and MRR 0.918; GateOnly has uniform role weights and observed MRR 1.000. On mixed FB15k-237 inputs, CCSC assigns more mass to correct roles than NoConf yet has lower MRR. These examples compare attention allocation with retrieval performance. The separately fitted projections prevent this comparison from isolating concentration as the cause of the differences.

Table 11: Low-temperature development diagnostics: fixed graph inputs at epoch 30 from models trained at role temperature 0.1, separate from the central-setting final evaluation. Clean means all roles correct; mixed contains both correct and wrong roles. Correct mass is the attention weight assigned to correct roles. Values are stored backbone-hierarchy means. These examples illustrate different orderings of attention and retrieval; all conditions are retained in the artifact. They are not a new confirmation or a causal intervention.

Panel / group / method	MRR	Mean max α	$\max \alpha \geq .95$	Correct mass
CoDEx-M / clean / CCSC	0.9177	0.8750	0.6331	1.0000
CoDEx-M / clean / GateOnly	1.0000	0.3333	0.0000	1.0000
FB15k-237 / mixed / CCSC	0.6980	0.7642	0.4841	0.9953
FB15k-237 / mixed / NoConf	0.7345	0.4420	0.0000	0.5597
WN18RR / mixed / CCSC	0.7918	0.9040	0.7006	0.8530
WN18RR / mixed / NoConf	0.7929	0.4951	0.0014	0.5803

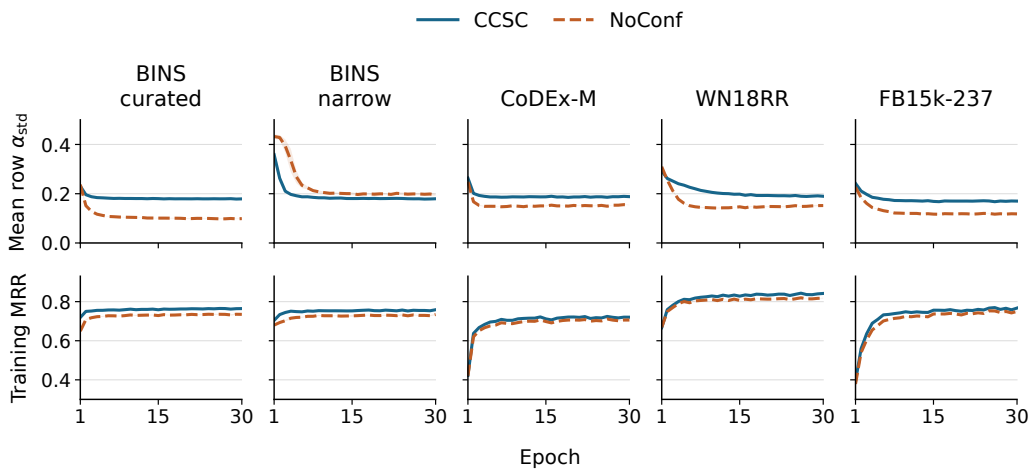

Figure 5: Learning at role temperature 0.5. Top: mean per-row population SD of the three attention weights, restricted to at least two present roles. Bottom: same-forward training MRR over all rows. Measurements precede the update and observations change each epoch. Shading is saved SD across five BINS repeats or three graph backbone means. Graph curves come from instrumented repeat runs.

F.2 Evaluation-only and matched-training temperatures

Figure 6's evaluation-only study fixes the checkpoint and tests ten temperatures from 0.1 through 1.0. Separately trained 0.1 and 1.0 models inherit the central recipe's learning-rate choices; they are not independently re-tuned at every temperature. The central 0.5 models are reused. The confidence and content terms both scale with temperature, and the entropy objective changes its operating point as well.

At matched training and evaluation temperature 1.0, CCSC has higher mean MRR than NoConf on all five panels (Table 12). At 0.1, that comparison reverses on WN18RR and FB15k-237; the largest negative CCSC−GateOnly difference at this temperature occurs on CoDEx-M. Thus, the gains over NoConf extend to 1.0, while the ordering of the fitted procedures depends on the temperature and panel.

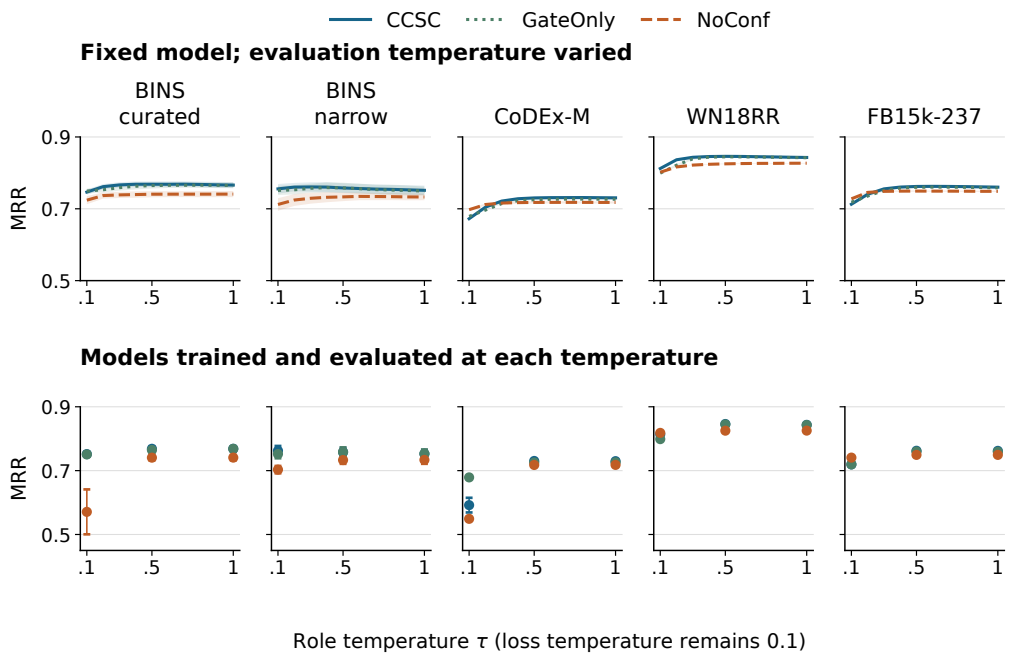

Figure 6: Temperature sensitivity on all five panels. Top: ten evaluation temperatures for fixed states trained at 0.5. Bottom: the three matched train/evaluation anchors; only 0.1 and 1.0 add from-scratch fits. Means and SD follow the declared BINS and graph repeat units. The loss temperature stays 0.1. These curves do not establish optimality of the central role temperature or isolate confidence amplification from content sharpness.

Table 12: Separately fitted CCSC, GateOnly, and NoConf at matched train/evaluation temperatures. The central 0.5 state is reused; 0.1 and 1.0 use from-scratch training with inherited central learning-rate choices. The intermediate evaluation-only grid is a different study. Mean ± SD uses the declared panel-specific repeats.

Panel / $\tau_{\text{train}} = \tau_{\text{eval}}$	CCSC − GateOnly	CCSC − NoConf
BINS curated / 0.1	$+0.0010 \pm 0.0017$	$+0.1809 \pm 0.0736$
BINS curated / 0.5	$+0.0038 \pm 0.0015$	$+0.0278 \pm 0.0028$
BINS curated / 1.0	$+0.0002 \pm 0.0011$	$+0.0276 \pm 0.0016$
BINS narrow / 0.1	$+0.0103 \pm 0.0023$	$+0.0592 \pm 0.0072$
BINS narrow / 0.5	-0.0002 ± 0.0017	$+0.0249 \pm 0.0027$
BINS narrow / 1.0	-0.0006 ± 0.0016	$+0.0189 \pm 0.0025$
CoDEx-M / 0.1	-0.0869 ± 0.0232	$+0.0429 \pm 0.0247$
CoDEx-M / 0.5	$+0.0052 \pm 0.0013$	$+0.0126 \pm 0.0009$
CoDEx-M / 1.0	$+0.0039 \pm 0.0013$	$+0.0117 \pm 0.0011$
WN18RR / 0.1	$+0.0153 \pm 0.0013$	-0.0042 ± 0.0006
WN18RR / 0.5	$+0.0014 \pm 0.0014$	$+0.0209 \pm 0.0008$
WN18RR / 1.0	$+0.0007 \pm 0.0010$	$+0.0180 \pm 0.0001$
FB15k-237 / 0.1	-0.0001 ± 0.0019	-0.0213 ± 0.0011
FB15k-237 / 0.5	$+0.0024 \pm 0.0013$	$+0.0129 \pm 0.0007$
FB15k-237 / 1.0	$+0.0030 \pm 0.0008$	$+0.0126 \pm 0.0002$

F.3 CONFIDENCE TRANSFORMATIONS

Confidence-quality transformations preserve the original missing-role mask and operate on present confidences. Squaring and square roots preserve their ordering; clipping can introduce ties. The full within-role permutation changes confidence–correctness association and is distinct from shuffling only within correctness buckets.

The confidence-OFF reference here sets present-role confidence to one. Appendix D explains its equivalence conditions and recorded-MRR agreement with omitting the log-confidence term. In Figure 7, full within-role permutation gives lower mean CCSC MRR than confidence OFF on all five panels.

Quality and retrieval across five illustrative coordinates. Table 13 uses the first listed repeat of each panel: repeat 42 for BINS, and backbone 13/Stage-2 seed 61001 for each graph. These post-hoc examples preserve the original CoDEx-M case and add the corresponding coordinates from the other panels. Within each coordinate, input, checkpoint, candidate, and target identities match across identity, square, and confidence OFF. Retrieval uses 2,669/1,497 query rows for BINS and 8,192 observations with 2,048 target anchors for each graph. Slot-quality diagnostics use only present slots within each role; their counts appear in the table. Expected calibration error (ECE) uses ten fixed bins, with empty bins retained as unavailable; Brier measures slot correctness and AUROC is tie-aware. Correct slots retain confidence one and occupy a different ECE bin from incorrect slots both before and after squaring. Incorrect confidence lies in $[0.3, 0.6]$, so within each incorrect-only bin the ECE contribution is its confidence sum divided by the present-slot count N. Let $y_i \in \{0, 1\}$ indicate whether present slot i is correct. Hence $\mathrm{ECE}(\gamma^2) = N^{-1} \sum_{i:y_i=0} \gamma_i^2 = \mathrm{Brier}(\gamma)$ in exact arithmetic. Brier becomes $N^{-1} \sum_{i:y_i=0} \gamma_i^4$ after squaring; AUROC remains one where both classes occur. Rounded displays and floating-point values need not be bit-identical. Original-confidence quality is input-dependent, so repeated Stage-2 models using the same input bank are not independent calibration samples. In the retained CoDEx-M example, MRR falls from 0.730093 to 0.713157, below confidence OFF (0.720411), while Brier/ECE decrease and AUROC stays one. The other illustrative coordinates retain their own values in Table 13. Squaring changes score scale and attention weights, so the retrieval change cannot be attributed to calibration alone. These are illustrative examples; the full map and reference results give the tested scope, not a new estimate of its frequency.

F.4 ADDITIONAL LOW-TEMPERATURE TRAJECTORIES

Figure 8 records changing training observations before each update. Its NoConf trajectories at role temperature 0.1 and the uniform replacements in Appendix D.1 refer to the same fitted models; the replacements evaluate terminal checkpoints on fixed inputs. The curated BINS NoConf family shows persistent near-one-hot weighting during learning. Figure 9 instead uses fixed graph development inputs after each epoch, with evolving model states.

Table 13: Confidence quality and retrieval at one illustrative coordinate per panel. BINS uses repeat 42; graphs use backbone 13 and Stage-2 seed 61001, the first registered coordinates. These are post-hoc examples, not panel averages or independent confirmation. Top: query-level MRR on the same observations for identity, square, and confidence OFF. Bottom: slot-level Brier/ECE on the same inputs, with present-slot counts N_s; AUROC is one for both signals in every row. Slot and query denominators differ. All coordinates remain in the supplied results.

Query retrieval

Dataset/tier	Query rows	Identity	Square	Confidence OFF
BINS curated	2,669	0.776361	0.768363	0.715194
BINS narrow	1,497	0.740900	0.739604	0.678263
CoDEx-M	8,192	0.730093	0.713157	0.720411
WN18RR	8,192	0.847733	0.842685	0.823730
FB15k-237	8,192	0.760779	0.745824	0.744387

Slot quality

			Brier		ECE		AUROC
Dataset/tier	Role	N_s	Identity	Square	Identity	Square	Both
BINS curated	A	2,242	0.0593	0.0128	0.1278	0.0593	1.0000
	O	2,271	0.0592	0.0123	0.1307	0.0592	1.0000
	T	2,279	0.0613	0.0128	0.1349	0.0613	1.0000
BINS narrow	A	1,270	0.0655	0.0138	0.1429	0.0655	1.0000
	O	1,283	0.0594	0.0117	0.1349	0.0594	1.0000
	T	1,283	0.0586	0.0121	0.1311	0.0586	1.0000
CoDEx-M	A	6,990	0.0794	0.0215	0.1535	0.0794	1.0000
	O	7,003	0.1019	0.0358	0.1722	0.1019	1.0000
	T	6,958	0.1036	0.0365	0.1744	0.1036	1.0000
WN18RR	A	6,998	0.0586	0.0121	0.1301	0.0586	1.0000
	O	6,977	0.1010	0.0343	0.1733	0.1010	1.0000
	T	6,940	0.0997	0.0339	0.1711	0.0997	1.0000
FB15k-237	A	6,913	0.0780	0.0210	0.1512	0.0780	1.0000
	O	6,996	0.1035	0.0359	0.1762	0.1035	1.0000
	T	6,924	0.1058	0.0373	0.1783	0.1058	1.0000

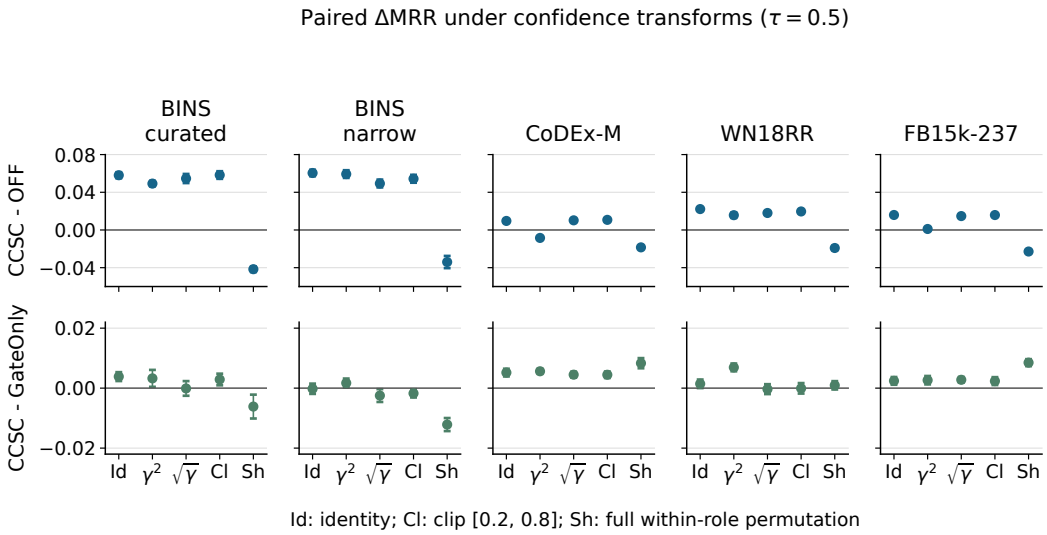

Figure 7: Confidence-quality interventions at role temperature 0.5. Top: transformed confidence in CCSC minus confidence OFF (present-role confidence set to one) in the same fitted state. Bottom: CCSC minus separately trained GateOnly under the same transform. Identity, square, square root, clipping to $[0.2, 0.8]$, and full within-role permutation are all retained. Bars are stored sample SD, not confidence intervals. The permutation is not restricted to correctness buckets.

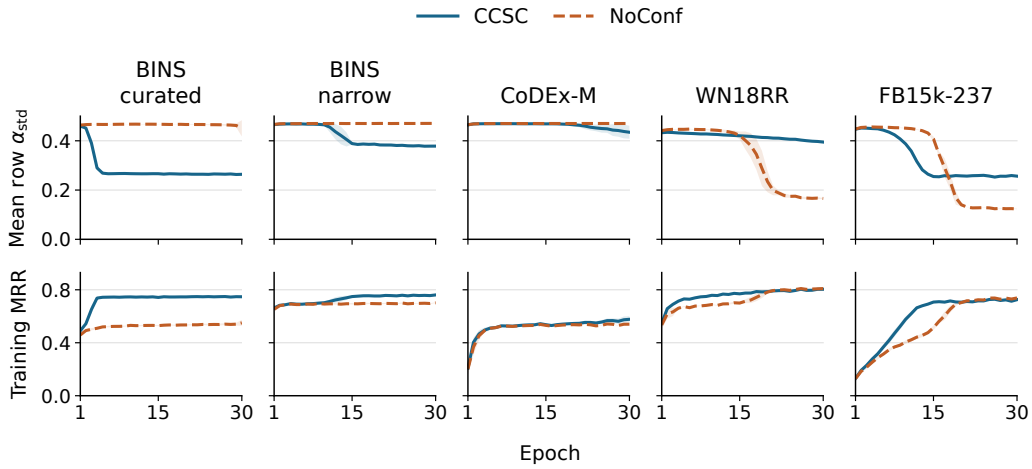

Figure 8: Attention and training retrieval at role temperature 0.1, from the matched-temperature study. The top row averages per-row population SD over three weights for rows with at least two present roles; bottom-row training MRR uses all rows. The timing, differing denominators, and stored SD convention match Figure 5. These curves come from models trained for the matched-temperature study.

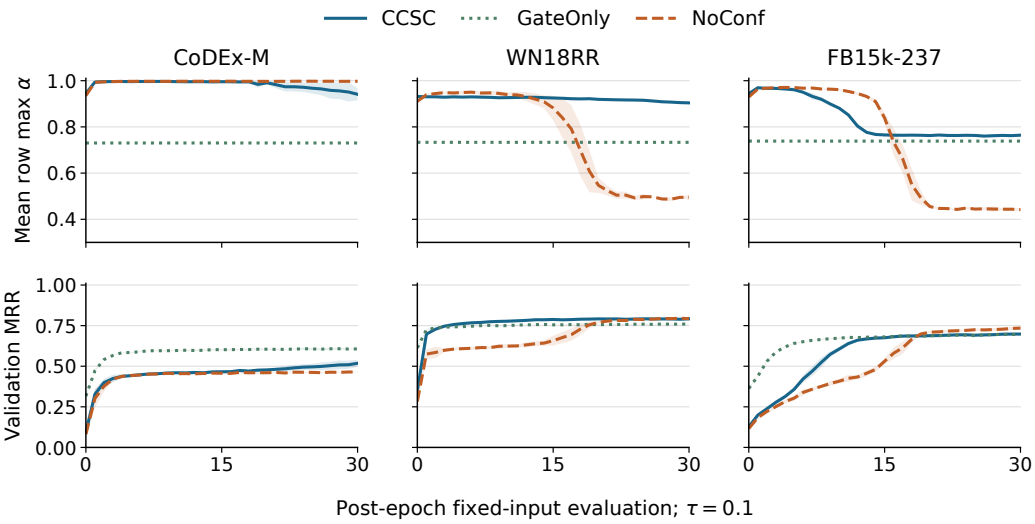

Figure 9: Fixed graph development inputs at role temperature 0.1, using the same mixed-correct/wrong mask for both plotted statistics. Epoch0 is the initial state; later points are post-epoch evaluation. Stored SD is across three backbone means. These are development measurements with evolving model and candidate representations, not confirmation or an intervention on concentration alone. BINS has no corresponding fixed-epoch validation in this runner.

G Artifact scope and bounded re-execution checks

Two corrected training runs were designated for reproduction before the new corrected results were available: CoDEx-M Uniform at backbone 71/Stage-2 seed 62017 and BINS curated SumMLP at repeat 42. The graph coordinate was a previously diagnosed nonfinite-state case. A separate operator reran both in a newly created environment on the same compute host. The compared model and optimizer states, used random states, and retained computational outputs matched with zero tolerance. The graph implementation did not use or seed NumPy; its captured NumPy state differed in all 31 checkpoints. This checks two specified computation paths, rather than an external implementation or the full experiment matrix. The additional check counts are retained in the accompanying numerical provenance notes.

The anonymous code artifact specifies which input panels and model states are included directly, the complete supported recipes and reference results, and the separate scope of package inference checks. Large preserved optimizer histories and full score matrices need not be duplicated in the review ZIP; their absence is stated explicitly. The artifact does not distribute private review material, credentials, or operational machine locations.