MERC: Co-evolutionary Multi-Geometry Message Passing for Zero-Shot Knowledge Graph Reasoning

Anonymous authors
Paper under double-blind review

Abstract

Structural knowledge graph foundation models enable zero-shot reasoning over unseen entities and relations by learning transferable structural patterns. However, most existing models rely on uniform entity-level message operators and largely static relation conditioning, which can impose fixed algebraic biases and limit adaptation to heterogeneous graph topologies. We propose MERC, a deterministic co-evolutionary multi-geometry structural KG reasoning model that couples parallel message passing in Complex, Split-Complex and Dual spaces with layer-wise entity-relation recalibration. The geometric branches provide complementary structural biases, while co-evolution dynamically adapts relation states using evolving entity contexts. Query-guided fusion then combines branch-specific evidence for each query. We evaluate MERC on 54 zero-shot KG benchmarks and 31 biomedical graphs. Results show that MERC improves average ranking performance over deterministic structural baselines, with clear gains under relation-vocabulary shift, compositional and antisymmetric patterns, and biomedical domain transfer. Ablation and efficiency analyses indicate that MERC's gains arise from the interaction between multi-geometry operators and relation recalibration, while MERC Flash offers a practical efficiency-oriented variant.

1 Introduction

Knowledge Graphs (KGs) represent factual knowledge as multi-relational structures, where each fact is typically encoded as a triple consisting of a head entity, a relation and a tail entity, such as (Paris, IsCapitalOf, France) (Hogan et al., 2021b). Knowledge Graph Completion (KGC), which aims to infer missing facts from incomplete KGs, has long been a central task in artificial intelligence (Liang et al., 2024; Nayyeri et al., 2021b). While conventional KGC models often rely on entity- and relation-specific embeddings, the fully inductive or zero-shot setting is substantially more challenging: the model must perform link prediction on new graphs whose entities and relation types may not have been observed during training. Recent structural Knowledge Graph Foundation Models (structural KGFMs) (Galkin et al., 2024; Liu et al., 2025), represented by Ultra (Galkin et al., 2024), address this setting by shifting the learning objective from memorizing graph-specific embeddings to learning transferable structural reasoning patterns. By constructing relation graphs (Lee et al., 2023; Galkin et al., 2024) and performing message passing over graph-agnostic meta-relations, these models can reason over unseen entities and relations without relying on predefined node features or textual descriptions.

Open Challenges. Despite their progress, existing structural KGFMs still face two important architectural bottlenecks. The first is *algebraic rigidity*. Many models rely on a single fixed relation transformation, commonly instantiated as a DistMult-style bilinear operator (Yang et al., 2015; Galkin et al., 2024), within the entity-level message passing function. This uniform algebraic design imposes a specific structural bias and may limit the model's ability to capture heterogeneous relational patterns, including symmetry, antisymmetry, inversion, hierarchy and composition (Nayyeri et al., 2021a). Element-wise multiplication is efficient and stable, but it is not equally well suited to asymmetric, hierarchical, compositional or one-to-many relational structures. The second bottleneck is *weakly adaptive relation conditioning*. Relation representations are often computed before entity-level inference and then used as fixed conditioning signals during message

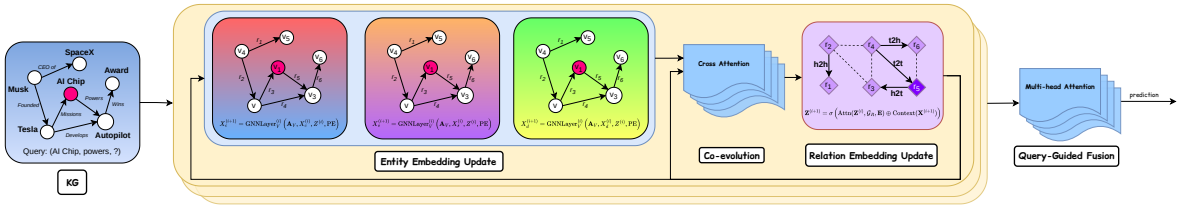


Figure 1: The overall framework of MERC. MERC performs multi-branch message passing using Complex, Split-Complex and Dual geometric operators to capture complementary relational patterns. Query-conditioned structural priors support entity propagation, while relation representations are refined over the marginalized relation graph and recalibrated using global statistical summaries of evolving entity states. A query-guided fusion module adaptively combines branch-specific outputs to produce the final prediction.

passing. This creates a largely one-directional dependency from relations to entities, although relation semantics are often shaped by the query-specific and entity-level topological contexts in which they appear. Relation graph meta-relations, such as head-to-head, tail-to-tail, head-to-tail and tail-to-head interactions, are therefore useful for zero-shot transfer but may remain insufficiently recalibrated according to the evolving entity-level reasoning state.

Recent models address parts of these limitations, but follow different design routes. Trix (Zhang et al., 2025) strengthens relation reasoning through fine-grained entity-indexed relation interaction, where relation-pair dependencies are distinguished by the specific entities through which they interact. This design is effective for preserving local relation-pair specificity, but it differs from operator-level geometric diversification and introduces additional graph-dependent relation structures. Flock (Kim et al., 2025) improves expressiveness through probabilistic equivariance, using stochastic random walks and recurrent sequence processors to break structural symmetries, at the cost of additional sampling and sequence-modeling overhead. These observations raise a central question: how can a structural KGFM improve algebraic expressiveness and entity-relation adaptivity while preserving deterministic and scalable inference?

Our Approach. We propose **MERC**, a **M**ulti-geometry **E**ntity-**R**elation **C**o-evolution architecture for zero-shot knowledge graph reasoning. MERC addresses the above limitations by replacing a uniform entity-level message operator with deterministic multi-geometry message passing. Specifically, MERC uses Complex (Trouillon et al., 2016), Split-Complex (Chami et al., 2020) and Dual (Cao et al., 2021) branches as complementary algebraic transformations inside a structural inference network. The Complex branch provides rotation- and phase-sensitive transformations that are useful for asymmetric, inverse and compositional patterns, the Split-Complex branch introduces hyperbolic-number-like interactions that can complement hierarchy-sensitive structures, and the Dual branch captures first-order and displacement-like relational corrections. These operators are used as transferable message functions rather than as standalone KGE scoring functions.

MERC further couples this multi-geometry propagation with adaptive relation conditioning. Instead of treating relation representations as fixed after relation graph inference, MERC recalibrates relation states across layers using global statistical summaries of evolving entity states. Unlike fine-grained entity-indexed relation interaction, MERC does not explicitly record which entity mediates each relation-pair dependency. It instead uses pooled entity context as a lightweight feedback signal, providing a different trade-off between local specificity and scalability. Query-guided fusion then adaptively combines branch-specific evidence according to the query relation, allowing the model to emphasize different geometric transformations for different relational patterns. MERC also incorporates query-conditioned structural priors and topology-aware relation graph refinement as supporting mechanisms. The central computation remains the coupling between multi-geometry entity propagation, layer-wise relation recalibration and query-guided branch fusion, as illustrated in Figure 1.

We evaluate MERC under a rigorous zero-shot setting against strong structural baselines, including Ultra (Galkin et al., 2024), Trix (Zhang et al., 2025) and Flock (Kim et al., 2025). The evaluation covers 54 diverse zero-shot KG benchmarks and 31 biomedical graphs designed to test cross-domain transfer under specialized graph topologies. In addition to standard ranking performance, we conduct relational-pattern analysis, systematic ablation studies and efficiency profiling to examine the roles of geometric branches, entity-relation co-evolution, query-guided fusion, structural priors and relation graph refinement.

Contributions:

- We introduce deterministic multi-geometry message passing for structural KGFMs, using Complex, Split-Complex and Dual operators as complementary branch-specific transformations for zero-shot knowledge graph reasoning.
- We design an adaptive entity-relation co-evolution and query-guided fusion framework that recalibrates relation states using evolving entity-level evidence and dynamically combines branch-specific representations according to the query relation.
- We provide a broad empirical evaluation on 54 zero-shot KG benchmarks and 31 biomedical graphs, including relational-pattern analysis, systematic ablations and efficiency profiling against deterministic and stochastic structural KGFM baselines.

2 Related Work

Algebraic and Geometric Operators for KG Reasoning. Knowledge graph representation learning has widely used algebraic operators to model relational patterns. Bilinear models such as RESCAL (Nickel et al., 2011) are expressive but parameter-intensive, while DistMult (Yang et al., 2015) improves scalability at the cost of symmetric scoring. Translational models, including TransE (Bordes et al., 2013) and its variants (Wang et al., 2014; Lin et al., 2015; Ji et al., 2015), represent relations as transformations between entities. More expressive operators further exploit non-real or non-Euclidean structures: ComplEx (Trouillon et al., 2016) and RotatE (Sun et al., 2019) model asymmetric, inverse and compositional relations through complex-valued interactions. QuatE (Zhang et al., 2019) and DualE (Cao et al., 2021) extend relation transformations to hypercomplex spaces. MuRP (Balazevic et al., 2019) and BoxE (Abboud et al., 2020) capture hierarchical or region-based relational structures. These methods show that different algebraic spaces encode different inductive biases. However, most of them are used as static embedding scorers tied to fixed entity and relation identifiers. In contrast, MERC uses Complex, Split-Complex and Dual transformations as transferable message functions inside a zero-shot structural KGFM.

Structural KGFMs and Zero-Shot KG Reasoning. Inductive KG reasoning methods such as GraIL (Teru et al., 2020), the Labeling Trick (Zhang et al., 2021), RED-GNN (Zhang & Yao, 2022), PathCon (Geng et al., 2023), AdaProp (Zhang et al., 2023) and A*Net (Zhu et al., 2023) exploit subgraphs, paths or adaptive propagation to generalize to unseen entities. NBFNet (Zhu et al., 2021) is particularly relevant because it formulates link prediction as query-conditioned Bellman-Ford-style path reasoning. Recent structural KGFMs further target zero-shot transfer across unseen relation vocabularies. InGram (Lee et al., 2023) and Ultra (Galkin et al., 2024) show that relation representations can be computed from graph-agnostic structural interactions, enabling cross-graph transfer without relation-id memorization. Theoretical studies on double equivariance, such as ISDEA and MTDEA (Zhou et al., 2025; 2023), further motivate models that remain equivariant to both entity and relation permutations. Recent extensions increase expressiveness through richer relation interactions, including MOTIF (Huang et al., 2025), GraphOracle (Du et al., 2025), Trix (Zhang et al., 2025) and Flock (Kim et al., 2025).

Relation Interaction, Co-evolution and Fusion. Ultra constructs a marginalized relation graph over four meta-relations and uses relation representations to guide entity-level inference, but its pipeline is largely relation-to-entity and typically uses a uniform message operator. Trix strengthens relation distinguishability through fine-grained entity-indexed relation interactions, preserving which entity mediates a relation-pair

interaction. Flock improves symmetry breaking through probabilistic double equivariance and controlled random walks. MERC differs from these directions by combining deterministic multi-geometry entity propagation with lightweight layer-wise relation recalibration. It does not construct an entity-indexed relation adjacency as in Trix. Instead, it summarizes evolving multi-branch entity states through statistical pooling inspired by PNA (Corso et al., 2020) and uses the resulting global context to recalibrate relation states across layers. Finally, query-guided attention fusion, related to attention-based graph learning (Veličković et al., 2017; Ying et al., 2021), adaptively combines branch-specific Complex, Split-Complex and Dual evidence according to the query relation.

3 Preliminaries

Knowledge Graphs. A knowledge graph is a directed multi-relational graph $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of entities, $\mathcal{R}$ denotes the set of relation types, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ is the set of factual triples (Hogan et al., 2021a). A triple $(h, r, t) \in \mathcal{E}$ represents a directed fact from a head entity $h \in \mathcal{V}$ to a tail entity $t \in \mathcal{V}$ under relation $r \in \mathcal{R}$.

Zero-Shot Link Prediction. Link prediction aims to infer missing facts in an incomplete knowledge graph. Given a query $(h, r, ?)$ or $(?, r, t)$, the model ranks candidate entities in $\mathcal{V}$ and assigns a score to each candidate entity. In this work, we focus on the fully inductive zero-shot setting, where inference is performed on graphs whose entity and relation vocabularies may be disjoint from those observed during training.

Let $\mathcal{G}_{\text{train}} = (\mathcal{V}_{\text{train}}, \mathcal{R}_{\text{train}}, \mathcal{E}_{\text{train}})$ denote the graph or collection of graphs used for training, and let $\mathcal{G}_{\text{inf}} = (\mathcal{V}_{\text{inf}}, \mathcal{R}_{\text{inf}}, \mathcal{E}_{\text{inf}})$ denote the graph used for inference. In the most challenging zero-shot setting, both the entity and relation vocabularies are disjoint:

$$\mathcal{V}_{\text{train}} \cap \mathcal{V}_{\text{inf}} = \emptyset, \qquad \mathcal{R}_{\text{train}} \cap \mathcal{R}_{\text{inf}} = \emptyset. \tag{1}$$

Therefore, the model cannot rely on learned lookup tables tied to specific entity or relation identifiers. Instead, it must learn structural reasoning rules that transfer across graph vocabularies.

Structural Relation Graphs. A central mechanism in recent zero-shot structural KG models is the construction of a relation graph. Given a knowledge graph $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$, the relation graph is defined as $\mathcal{G}_r = (\mathcal{R}, \mathcal{R}_{\text{fund}}, \mathcal{E}_r)$, where nodes correspond to relation types in the original graph and edges encode graph-agnostic interactions among relations. Following Ultra (Galkin et al., 2024), the relation graph is built from four fundamental meta-relations:

$$\mathcal{R}_{\text{fund}} = \{\text{h2h}, \text{t2t}, \text{h2t}, \text{t2h}\}. \tag{2}$$

These meta-relations indicate whether two relations share head entities, tail entities, or head-tail endpoints. For example, a tail-to-head interaction captures the case where the tail of one relation can serve as the head of another, which is important for compositional reasoning.

The relation graph enables zero-shot transfer because it represents relations through their structural interactions rather than through their identifiers. In Ultra (Galkin et al., 2024), a relation-level message passing network first computes query-conditioned relation representations over $\mathcal{G}_r$. These relation representations are then used as edge features in an entity-level message passing network over the original graph $\mathcal{G}$. MERC builds on this structural formulation, while Section 4 describes how relation states and entity states are further coupled during inference.

Query-Conditioned Message Passing. The entity-level reasoning backbone of many structural KG models is query-conditioned message passing. A generic relational message passing layer updates the representation of an entity v by aggregating messages from its incoming neighbors:

$$\mathbf{h}_v^{(l+1)} = \text{UPDATE}\left(\mathbf{h}_v^{(l)}, \text{AGG}\left(\left\{\text{MSG}\left(\mathbf{h}_u^{(l)}, \mathbf{r}_r^{(l)}\right) : (u, r, v) \in \mathcal{E}\right\}\right)\right), \tag{3}$$

where $\mathbf{h}_v^{(l)}$ is the representation of entity v at layer l, $\mathbf{r}_r^{(l)}$ is the representation of relation r at layer l, and AGG is a permutation-invariant aggregation operator.

NBFNet (Zhu et al., 2021) provides a path-based interpretation of this process by neuralizing the Bellman-Ford algorithm. For a query $(h, q, ?)$, the source entity h is initialized with a query-dependent boundary condition:

$$\mathbf{h}_v^{(0)} = \mathbb{I}(v = h)\mathbf{q}, \tag{4}$$

where $\mathbf{q}$ denotes the query relation representation and $\mathbb{I}(\cdot)$ is the indicator function. Repeated message passing then propagates query-conditioned information through paths starting from the source entity. After L layers, candidate tail entities are scored using their final representations.

Ultra (Galkin et al., 2024) extends this formulation to the fully inductive setting by replacing fixed relation embeddings with relation representations computed from the relation graph. Section 4 introduces how MERC instantiates the generic message function with multiple geometric operators and updates relation states together with evolving entity contexts.

4 Methodology

MERC implements the layer-wise co-evolutionary multi-geometry framework introduced above as a structural reasoning model. Given a knowledge graph $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$, its marginalized relation graph $\mathcal{G}_r = (\mathcal{R}, \mathcal{R}_{\text{fund}}, \mathcal{E}_r)$, and a batch of entity prediction queries, MERC alternates between branch-separated entity propagation and relation-state recalibration. The central computation consists of three coupled operations: multi-geometry entity message passing, relation recalibration from evolving entity states and query-guided branch fusion.

4.1 Layer-wise MERC Reasoning

Let B be the batch size, $N = |\mathcal{V}|$, $R = |\mathcal{R}|$, and L the number of propagation layers. MERC uses the branch set

$$\mathcal{M} = \{\text{Complex}, \text{Split-Complex}, \text{Dual}\}. \tag{5}$$

For each query $(h_b, r_b, ?)$, MERC initializes the queried relation and the source entity with a shared query signal. It then adds a gated shortest-path structural prior to the initial entity states. At each layer, every branch propagates entity states over the same graph using the current relation states and a branch-specific geometric operator. The updated branch states are summarized into statistical entity-context tokens, which are then used to recalibrate relation states for the next layer. After L layers, MERC uses the query relation state to guide branch fusion and score candidate entities.

This order follows the co-evolution design: relation states guide the current entity update, and the resulting entity states provide feedback for the next relation update. MERC therefore does not construct entity-indexed relation interactions. It operates on a marginalized relation graph and uses global entity-context feedback as a scalable relation recalibration signal.

4.2 Query-Conditioned Initialization

We denote the batch-wise relation state tensor by $\mathbf{X}^{(l)}$, where the representation of relation r for query b is $\mathbf{x}_{b,r}^{(l)}$. For a query batch, MERC initializes relation states as

$$\mathbf{X}_{b,r}^{(0)} = \mathbb{I}(r = r_b)\mathbf{1}, \qquad \mathbf{X}^{(0)} \in \mathbb{R}^{B \times R \times d}, \tag{6}$$

where $\mathbf{1} \in \mathbb{R}^d$ is a constant query signal. The initial query representation is

$$\mathbf{q}_b = \mathbf{X}_{b,r_b}^{(0)}. \tag{7}$$

Each branch uses the same source boundary:

$$\mathbf{H}_{m,b,j}^{(0)} = \mathbb{I}(j = h_b)\mathbf{q}_b, \qquad m \in \mathcal{M}. \tag{8}$$

MERC further injects a query-relative structural prior using bounded shortest-path distance (SPD). For each source h_b, a batched BFS computes a truncated distance matrix $\mathbf{D} \in \mathbb{R}^{B\times N}$ with maximum distance $d_{\max} = L + 1$. Distances are embedded as

$$\mathbf{P} = \mathbf{E}_{\text{dist}}(\mathbf{D}) \in \mathbb{R}^{B\times N\times d_l}. \tag{9}$$

The positional signal is scaled by a relation-frequency gate:

$$g_{r_b} = \sigma\left(\frac{\mathbf{W}_{\text{gate}}[\mathbf{q}_b \parallel f_{r_b}] + b_{\text{gate}}}{\tau}\right), \qquad \tilde{\mathbf{P}}_{b,j} = g_{r_b}\mathbf{P}_{b,j}. \tag{10}$$

Here f_{r_b} is computed from the current inference graph as the normalized frequency of relation r_b, so it does not depend on training-time relation identities. The final initial state is

$$\mathbf{H}^{(0)}_{m,b,j} = \mathbb{I}(j = h_b)\mathbf{q}_b + \tilde{\mathbf{P}}_{b,j}. \tag{11}$$

4.3 Multi-Geometry Entity Propagation

At layer l, all branches receive the same graph, query boundary and relation-state tensor $\mathbf{X}^{(l)}$, but use different message functions. Let $\mathbf{H}^{(l)}_m \in \mathbb{R}^{B\times N\times d_l}$ denote entity states in branch m. Each branch internally projects the shared relation states into its message space:

$$\mathbf{R}^{(l)}_m = \Pi_m(\mathbf{X}^{(l)}). \tag{12}$$

For edge $(u, r, v) \in \mathcal{E}$, the message in branch m is

$$\mathbf{m}^{(l,m)}_{u,r,v} = \mu_m\left(\mathbf{h}^{(l)}_{m,u}, \mathbf{r}^{(l)}_{m,r}\right), \tag{13}$$

and the aggregated message is

$$\mathbf{a}^{(l,m)}_v = \text{AGG}\left(\mathbf{m}^{(l,m)}_{u,r,v} : (u, r, v) \in \mathcal{E}\right). \tag{14}$$

The branch-specific entity state is updated by

$$\mathbf{h}^{(l+1)}_{m,v} = \text{LayerNorm}\left(\sigma\left(\mathbf{W}^{(l)}_m[\mathbf{h}^{(l)}_{m,v} \parallel \mathbf{a}^{(l,m)}_v]\right) + \mathbf{h}^{(l)}_{m,v}\right), \tag{15}$$

where the residual connection is applied.

For all geometric branches, entity and relation vectors are split into two components,

$$\mathbf{h} = \mathbf{h}_{\text{re}} \parallel \mathbf{h}_{\text{im}}, \qquad \mathbf{r} = \mathbf{r}_{\text{re}} \parallel \mathbf{r}_{\text{im}}. \tag{16}$$

The branch operators are

$$\begin{aligned}
\mu_{\text{Complex}}(\mathbf{h}, \mathbf{r}) &= (\mathbf{h}_{\text{re}} \circ \mathbf{r}_{\text{re}} - \mathbf{h}_{\text{im}} \circ \mathbf{r}_{\text{im}}) \parallel (\mathbf{h}_{\text{re}} \circ \mathbf{r}_{\text{im}} + \mathbf{h}_{\text{im}} \circ \mathbf{r}_{\text{re}}),\\
\mu_{\text{Split-Complex}}(\mathbf{h}, \mathbf{r}) &= (\mathbf{h}_{\text{re}} \circ \mathbf{r}_{\text{re}} + \mathbf{h}_{\text{im}} \circ \mathbf{r}_{\text{im}}) \parallel (\mathbf{h}_{\text{re}} \circ \mathbf{r}_{\text{im}} + \mathbf{h}_{\text{im}} \circ \mathbf{r}_{\text{re}}),\\
\mu_{\text{Dual}}(\mathbf{h}, \mathbf{r}) &= (\mathbf{h}_{\text{re}} \circ \mathbf{r}_{\text{re}}) \parallel (\mathbf{h}_{\text{re}} \circ \mathbf{r}_{\text{im}} + \mathbf{h}_{\text{im}} \circ \mathbf{r}_{\text{re}}).
\end{aligned} \tag{17}$$

The branches remain separated throughout propagation and are combined only during final query-guided fusion.

4.4 Relation Recalibration from Entity Context

After entity propagation at layer l, MERC concatenates branch states:

$$\mathbf{C}^{(l)} = \text{Concat}_{m\in\mathcal{M}}\left(\mathbf{H}^{(l+1)}_m\right) \in \mathbb{R}^{B\times N\times d_{\text{ctx}}}. \tag{18}$$

It summarizes the entity states into four statistical tokens:

$$\mathbf{E}_{\text{ctx}}^{(l)} = \text{Concat}_{\text{token}}\left(\mathbf{c}_{\text{mean}}^{(l)}, \mathbf{c}_{\text{max}}^{(l)}, \mathbf{c}_{\text{min}}^{(l)}, \mathbf{c}_{\text{std}}^{(l)}\right) \in \mathbb{R}^{B\times 4\times d_{\text{ctx}}}, \tag{19}$$

where the four tokens are computed over the entity dimension. These tokens provide compact feedback from the evolving entity-level reasoning state.

Relation states are first refined over the marginalized relation graph. For a meta-edge $(i, t_{ij}, j) \in \mathcal{E}_r$, with $t_{ij} \in \text{h2h}, \text{t2t}, \text{h2t}, \text{t2h}$, MERC injects a meta-edge embedding into key and value projections:

$$\mathbf{k}'_{b,j} = \mathbf{W}_K \mathbf{x}_{b,j}^{(l)} + \mathbf{e}_{t_{ij}}, \qquad \mathbf{v}'_{b,j} = \mathbf{W}_V \mathbf{x}_{b,j}^{(l)} + \mathbf{e}_{t_{ij}}. \tag{20}$$

The relation graph attention is

$$\alpha_{b,ij} = \text{softmax}_{j\in\mathcal{N}_r(i)}\left(\frac{(\mathbf{W}_Q \mathbf{x}_{b,i}^{(l)})^\top \mathbf{k}'_{b,j}}{\sqrt{d_{\text{head}}}}\right), \tag{21}$$

where $d_{head} = d_l/H_{re}$, H_{re} denotes the number of relation graph attention heads, and the topology-refined relation state is

$$\tilde{\mathbf{x}}_{b,i}^{(l)} = \text{LayerNorm}\left(\mathbf{x}_{b,i}^{(l)} + \text{Dropout}\left(\mathbf{W}_O \sum_{j\in\mathcal{N}_r(i)} \alpha_{b,ij}\mathbf{v}'_{b,j}\right)\right). \tag{22}$$

The topology-refined relation states then attend to the four entity-context tokens:

$$\text{Att}_{\text{co}} = \text{softmax}\left(\frac{\mathbf{W}_{\text{co}}^Q \tilde{\mathbf{X}}^{(l)} (\mathbf{W}_{\text{co}}^K \mathbf{E}_{\text{ctx}}^{(l)})^\top}{\sqrt{d_k}}\right) \mathbf{W}_{\text{co}}^V \mathbf{E}_{\text{ctx}}^{(l)}. \tag{23}$$

where $d_k = d_l/H_{co}$, H_{co} denotes the number of co-evolution attention heads. The next-layer relation states are

$$\mathbf{X}^{(l+1)} = \text{LayerNorm}\left(\tilde{\mathbf{X}}^{(l)} + \text{Dropout}(\mathbf{W}_{\text{out}}\,\text{Att}_{\text{co}})\right). \tag{24}$$

This mechanism implements lightweight layer-wise relation recalibration without constructing entity-indexed relation-pair structures.

4.5 Query-Guided Fusion and Scoring

After L layers, MERC scores candidate entities using the final query relation state

$$\mathbf{q}_b^{\text{upd}} = \mathbf{X}_{b,r_b}^{(L)}. \tag{25}$$

If hidden concatenation is enabled, the branch representation for entity v is

$$\mathbf{z}_{b,v,m} = \text{Concat}_{l=1}^{L}\, \mathbf{h}_{b,v,m}^{(l)}; \tag{26}$$

otherwise $\mathbf{z}_{b,v,m} = \mathbf{h}_{b,v,m}^{(L)}$. Each branch is independently normalized as $\bar{\mathbf{z}}_{b,v,m}$.

For MERC Flash, which has a single branch, scoring bypasses attention:

$$\phi_b(v) = \text{MLP}_{\text{score}}\left([\bar{\mathbf{z}}_{b,v,1} \parallel \mathbf{q}_b^{\text{upd}}]\right). \tag{27}$$

For full MERC, the query relation attends over branch representations:

$$\alpha_{b,v,h,m} = \text{softmax}_m\left(\frac{\mathbf{Q}_{b,h}^\top \mathbf{K}_{b,v,m,h}}{\sqrt{d_h}}\right), \tag{28}$$

where $d_h = d_l/H_{fu}$, H_{fu} denotes the number of query-guided fusion attention heads, $\mathbf{Q}_{b,h}$ is projected from $\mathbf{q}_b^{\text{upd}}$, and $\mathbf{K}_{b,v,m,h}$ is projected from $\bar{\mathbf{z}}_{b,v,m}$. MERC reweights branch values and concatenates them:

$$\mathbf{o}_{b,v} = \text{Concat}_{m=1}^{M} \text{Concat}_{h=1}^{H} \alpha_{b,v,h,m} \mathbf{V}_{b,v,m,h}. \tag{29}$$

The final score is

$$\phi_b(v) = \text{MLP}_{\text{score}} \left([\mathbf{o}_{b,v} \parallel \mathbf{q}_b^{\text{upd}}] \right). \tag{30}$$

Thus, fusion is query-guided and candidate-specific, while branch-separated evidence is preserved for the final scorer.

4.6 Training Objective and Model Analysis

MERC follows the base structural NBFNet-style link prediction objective using candidate scores $\phi_b(v)$. For multi-branch MERC, an entropy bonus is applied to the query-guided branch attention weights:

$$\mathcal{L} = \mathcal{L}_{\text{link}} - \lambda \mathcal{H}(\boldsymbol{\alpha}). \tag{31}$$

Since the entropy term is subtracted from the training objective, it encourages broader branch usage and mitigates premature collapse to a single branch. This auxiliary term is not used for the single-branch MERC Flash variant.

Computational efficiency. MERC is designed to improve expressiveness without relying on stochastic walk sampling, recurrent sequence processing or entity-indexed relation-pair materialization. Its relation-state refinement operates on the marginalized relation graph $\mathcal{G}_r$, while entity-to-relation feedback is provided through four global statistical context tokens. As a result, MERC avoids constructing relation interactions separately for each mediating entity. If α denotes the maximum number of unique relations incident to an entity, entity-indexed relation interaction can introduce a graph-dependent term of order $\mathcal{O}(|\mathcal{V}|\alpha^2)$. MERC instead uses a marginalized relation graph term bounded by $\mathcal{O}(|\mathcal{R}|^2)$, together with linear entity-level propagation and pooling terms. Under fixed branch number, depth and hidden dimension, MERC's dominant graph-size-dependent cost can be summarized as

$$\mathcal{O}(|\mathcal{E}| + |\mathcal{V}| + |\mathcal{R}|^2), \tag{32}$$

with memory scaling linearly in the number of entities. The full complexity, including bounded SPD computation, multi-branch propagation, relation-to-context attention and candidate fusion, is provided in Appendix A.

Expressivity. MERC and entity-indexed relation interaction provide complementary forms of expressiveness. Entity-indexed methods preserve fine-grained local relation-pair specificity by recording which entity mediates a relation interaction. MERC instead enriches the operator space of structural reasoning: Complex, Split-Complex and Dual branches propagate entity evidence under different algebraic biases, while layer-wise relation recalibration adapts relation states to the evolving entity-level reasoning context. Query-guided fusion further lets the model combine branch-specific evidence according to the query relation. Therefore, MERC does not claim to strictly dominate fine-grained entity-indexed interaction in all settings. Rather, it offers a deterministic and scalable alternative that couples multi-geometry message passing with lightweight entity-relation co-evolution. A more detailed expressivity discussion is deferred to Appendix B.

5 Experiments

We evaluate MERC under strict zero-shot link prediction across diverse knowledge graphs. The experiments are designed to answer four research questions that follow the co-evolutionary multi-geometry formulation introduced in Section 4. **RQ1**: Does co-evolutionary multi-geometry reasoning improve zero-shot link prediction across heterogeneous unseen knowledge graphs? **RQ2**: How does MERC behave across relational patterns and domain-shifted biomedical knowledge graphs? **RQ3**: What accuracy-efficiency trade-offs do MERC Flash and full MERC exhibit? **RQ4**: How do geometric operators, layer-wise relation recalibration, structural priors, topology-aware relation refinement and query-guided fusion contribute to MERC's zero-shot transfer ability?

5.1 Overview

Model variants and baselines. We evaluate two variants of MERC. **MERC Flash** uses a single `Complex` branch and serves as the efficiency-oriented variant. **MERC** uses three branches, namely `Complex`, `Split-Complex` and `Dual`, and represents the full co-evolutionary multi-geometry architecture. The main deterministic structural baselines are Ultra (Galkin et al., 2024) and Trix (Zhang et al., 2025). Flock (Kim et al., 2025) is additionally included in relational-pattern, biomedical and efficiency comparisons, where its stochastic random-walk design provides a useful reference for expressiveness and scalability.

MERC is pre-trained on FB15k-237 (Toutanova & Chen, 2015), WN18RR (Dettmers et al., 2018) and CoDEx-M (Safavi & Koutra, 2020). Zero-shot evaluation is performed directly on unseen target graphs without target-specific training or fine-tuning. The benchmark suite contains 57 graphs, among which 54 are used for zero-shot evaluation, covering Inductive e, r, Inductive e and transductive settings. Following prior practice (Lv et al., 2020; Galkin et al., 2024; Zhang et al., 2025), only tail prediction is evaluated on FB15k-237-10%, FB15k-237-20% and FB15k-237-50% (Lv et al., 2020).

Evaluation and statistical testing. We report standard filtered ranking metrics, including MRR, MR and Hits@K where applicable. Since zero-shot inference is deterministic given a trained checkpoint, each evaluation is executed once following prior structural KGFM practice (Galkin et al., 2024). Statistical tests reported below are paired Wilcoxon signed-rank tests (Wilcoxon, 1945) over datasets or pattern-specific subsets rather than repeated stochastic runs. These tests therefore measure cross-graph or cross-subset consistency, not run-to-run training variance. Full training configurations, hardware settings, hyperparameters and checkpoint-selection rules are reported in the Appendix D.

Fairness and budget control. All deterministic models are evaluated under the same zero-shot protocol: the same pre-training graphs, no target-graph fine-tuning, the same filtered ranking evaluation and the same target splits. For efficiency comparisons, we additionally report parameter count, training-time overhead, inference latency and peak GPU memory under the same hardware setting. When a model cannot run under the unified memory budget, we report the failure rather than excluding the case. The parameter counts are 168,705 for Ultra, 87,138 for Trix, 801,969 for Flock, 81,188 for MERC Flash and 176,548 for MERC. Thus, full MERC has a parameter budget comparable to Ultra and far smaller than Flock, while MERC Flash is slightly smaller than Trix.

5.2 Comprehensive Zero-Shot Link Prediction

Setup. This experiment evaluates whether MERC improves general zero-shot link prediction across heterogeneous graph domains. We compare Ultra, Trix, MERC Flash and MERC on 54 unseen graphs grouped into Inductive e, r, Inductive e and transductive settings. Detailed information on the three pre-training graphs and the 54 unseen zero-shot evaluation graphs is provided in Appendix E. This setup tests two questions: whether replacing a uniform real-valued message operator with a geometry-aware operator improves structural transfer, and whether the full multi-geometry design provides additional gains beyond the single-branch MERC Flash variant. The comparison with Trix is especially important because Trix uses fine-grained entity-indexed relation interaction, whereas MERC uses marginalized relation graph refinement and pooled entity-context recalibration.

Results. Table 1 shows that MERC achieves the best overall average and the strongest performance in the Inductive e, r and transductive settings. This supports the claim that co-evolutionary multi-geometry reasoning is useful under stronger target-graph distribution shift, especially when both entities and relations in the test triples are less directly supported by the observed graph, and relation behavior must be inferred from structural context. MERC Flash remains competitive despite using only a single `Complex` branch, indicating that geometry-aware message functions already provide a strong transferable inductive bias. Detailed dataset-level results for all 54 unseen benchmark graphs are reported in Appendix J.

The scenario-level results also clarify the boundary between MERC and Trix. Trix remains slightly stronger in the Inductive e setting, where relation types are shared and the main distribution shift concerns entities.

Model	Inductive e, r (23 graphs)		Inductive e (18 graphs)		Transductive (13 graphs)		Total Avg (54 graphs)	
	MRR	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**
Ultra	0.345	0.513	0.431	0.556	0.312	0.448	0.366	0.518
Trix	0.368	0.540	**0.455**	**0.592**	0.339	0.500	0.390	0.548
MERC Flash	0.379	0.546	0.444	0.591	0.339	0.495	0.391	0.549
MERC	**0.391**	**0.556**	0.446	0.588	**0.347**	**0.509**	**0.398**	**0.555**

Table 1: Average zero-shot link prediction results (MRR and Hits@10) of Ultra, Trix and MERC family, grouped by different scenarios over 54 KGs.

This suggests that Trix's fine-grained entity-indexed relation interaction can be particularly useful when the relation space is shared and the shift is mainly over entities. In contrast, MERC provides stronger aggregate transfer under relation vocabulary shift and transductive structural transfer, which is consistent with its design choice of combining scalable relation recalibration with complementary geometric message operators.

Takeaway (RQ1)

Across 54 unseen knowledge graphs, MERC achieves the strongest overall zero-shot performance, with the clearest gains under relation vocabulary shift and transductive structural transfer. MERC Flash shows that a single hypercomplex operator already provides a strong transferable inductive bias, while full MERC further improves robustness through coupled multi-geometry propagation and layer-wise relation recalibration.

5.3 Relational Pattern and Cross-Domain Transfer

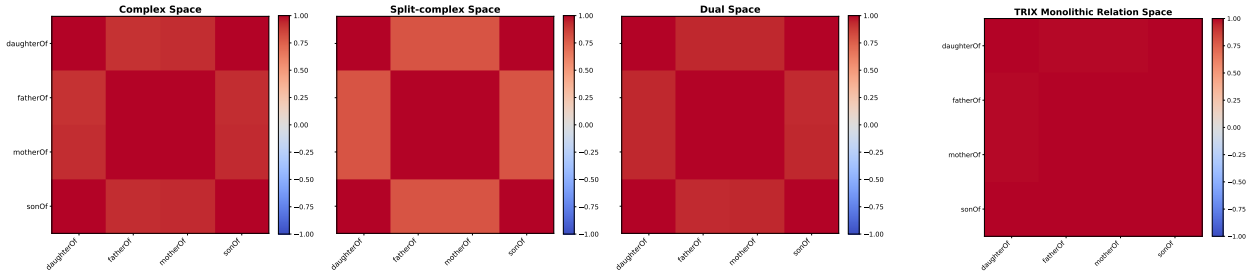
Setup. This section evaluates whether MERC's gains are associated with specific structural reasoning patterns and whether they transfer to domains beyond standard encyclopedic KGs. We first extract relational pattern subsets from the 54 benchmark graphs using confidence-based pattern identification following (Ali et al., 2022). The exact definitions, confidence threshold, mining procedure, and pattern-specific test subset construction are reported in Appendix F. Test triples are grouped into symmetry, antisymmetry and composition categories. All models are evaluated on the same fixed subsets. We report micro-averaged MRR and Hits@10 over triples within each pattern category to reduce instability from sparse subsets. Paired Wilcoxon signed-rank tests (Wilcoxon, 1945) are conducted across pattern-specific subsets.

We then examine domain and representation behavior from three complementary angles. First, MetaFam (Zhou et al., 2023) is used as a case study for conflicting and compositional family relations, including `fatherOf`, `motherOf`, `parentOf`, `childOf`, `sonOf` and `daughterOf`. These relations are useful for qualitative error analysis because they involve directionality, inverse constraints and semantic overlap. Second, 31 biomedical Protein-Disease graphs reconstructed from Clinical Knowledge Graph (CKG) resources are used to test transfer under strong domain shift, skewed degree distributions and specialized biological topologies. The construction procedure, data provenance, splitting strategy and graph-level statistics of the 31 biomedical Protein-Disease graphs are provided in Appendix G. Third, relation embedding visualizations on MetaFam and NELL-995 (Xiong et al., 2017) are used as qualitative supporting evidence for relation separation. These visualizations are not treated as independent proof of representation disentanglement. The complete NELL-995 relation similarity heatmaps comparing MERC and Trix are provided in Appendix H.

Results. Table 2 provides a finer-grained view of the structural capabilities behind the overall benchmark results. MERC shows its clearest advantage on antisymmetric relations when it improves both MRR and Hits@10 over deterministic baselines. This pattern is important because directed relations are precisely where symmetric or weakly directional real-valued operators are most constrained. The result is consistent with the intended role of Complex and Dual transformations in improving direction-sensitive reasoning.

Model	Type	Composition (41 subsets)		Symmetry (20 subsets)		Antisymmetry (54 subsets)	
		MRR	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**
Ultra	Deterministic	0.341	0.489	0.805	0.882	0.289	0.430
Trix	Deterministic	0.337	0.494	0.846	**0.919**	0.268	0.431
Flock	Stochastic	**0.368**	**0.530**	0.849	0.916	0.299	0.448
MERC Flash	Deterministic	$0.359^{\dagger\dagger}$	$0.513^{\dagger\dagger}$	0.844	0.906	$0.292^{\dagger\dagger}$	$0.442^{\dagger\dagger}$
MERC	Deterministic	$0.364^{\dagger\dagger}$	$0.522^{\dagger\dagger}$	**0.865**	0.899	$\mathbf{0.309}^{\dagger\dagger\dagger\ddagger}$	$\mathbf{0.462}^{\dagger\dagger\dagger\ddagger}$

Table 2: Micro-averaged link prediction results (MRR and Hits@10) across different relational patterns. The best results among all models are highlighted in **bold**, and the second-best results are marked with an underline. † and ‡ indicate that our model significantly outperforms the major baselines Ultra and Trix ($p < 0.05$ for single symbol, $p < 0.01$ for double symbols) under a paired Wilcoxon signed-rank test conducted across relational-pattern subsets, respectively.



(a) MERC Subspaces (b) Trix Subspace

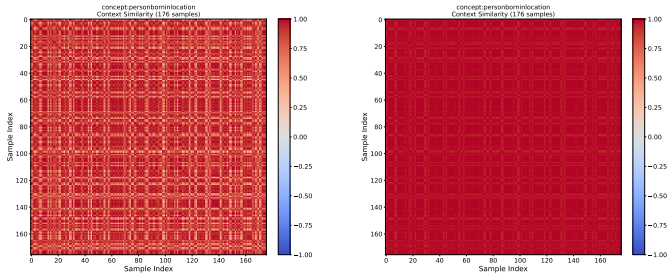
Figure 2: Inter-relational cosine similarity heatmaps on the MetaFam validation set.

On compositional subsets, MERC substantially improves over Ultra and Trix and approaches the stochastic Flock baseline while retaining deterministic inference. This indicates that deterministic multi-space propagation can capture a large portion of multi-hop relational structure without requiring random-walk sampling. Symmetry is comparatively saturated across models, so it offers less discriminative evidence for architectural differences.

The qualitative analyses in Figure 2 and Figure 3 are used as supporting evidence for the pattern-level results. On MetaFam, the analysis focuses on whether MERC reduces confusion among directionally or semantically related family relations, such as parent-child inverse errors. MERC shows clearer visual separation among

Model	Type	Protein-Disease biomedical Datasets (31 graphs)				
		MR	**MRR**	**Hits@1**	**Hits@3**	**Hits@10**
Ultra	Deterministic	3396.728	0.151	0.096	0.165	0.259
Trix	Deterministic	1523.252	0.196	0.131	0.237	0.320
Flock	Stochastic	**1496.708**	**0.256**	**0.187**	0.276	0.388
MERC Flash	Deterministic	$1497.293^{\dagger\dagger}$	$0.247^{\dagger\dagger\dagger\ddagger\ddagger **}$	$0.175^{\dagger\dagger\dagger\ddagger\ddagger **}$	$0.270^{\dagger\dagger\dagger\ddagger *}$	$0.388^{\dagger\dagger\dagger\ddagger **}$
MERC	Deterministic	$1557.484^{\dagger\dagger}$	$0.255^{\dagger\dagger\dagger\ddagger *}$	$0.185^{\dagger\dagger\dagger\ddagger}$	$\mathbf{0.277}^{\dagger\dagger\dagger\ddagger *}$	$\mathbf{0.392}^{\dagger\dagger\dagger\ddagger **}$

Table 3: Micro-averaged zero-shot link prediction results on the 31 Protein-Disease biomedical datasets. The best results are highlighted in **bold** and the second-best are underlined. † and ‡ indicate that the MERC framework significantly outperforms Ultra and Trix, respectively ($p < 0.01$ for double symbols). $*$ and $**$ indicate a statistically significant difference from Flock ($p < 0.05$ and $p < 0.01$, respectively), not necessarily an improvement over Flock, under a paired Wilcoxon signed-rank test conducted across 31 biomedical datasets.



(a) MERC (b) Trix

Figure 3: Comparison of cosine-similarity heatmaps for the relation *person born in location* during the zero-shot inference phase on the NELL-995 test set.

several conflicting family relations than the compared Trix representation. The NELL-995 visualization similarly illustrates MERC's finer relation separation during zero-shot inference.

Table 3 extends the comparison to biomedical Protein-Disease graphs, where the target domain differs substantially from the standard KG benchmarks. MERC achieves competitive MRR and strong top-K retrieval under biomedical domain shift, while MERC Flash offers a lighter deterministic configuration that may be preferable when biomedical graphs are sparse or inference efficiency is critical. These results show MERC learns transferable structural and relational biases rather than only fitting the topology of standard benchmark domains. Detailed dataset-level results for the relational pattern subsets and the biomedical Protein-Disease graphs are reported in Appendix K.

Takeaway (RQ2)

MERC's advantages are most evident in directed, compositional and domain-shifted reasoning scenarios. It improves antisymmetric and compositional pattern modeling over deterministic baselines and maintains strong transfer on 31 Protein-Disease biomedical graphs. These results indicate that MERC learns transferable relational biases rather than only exploiting benchmark-specific topology.

5.4 Efficiency vs. Performance Analysis

Setup. This experiment evaluates the scalability trade-off in **RQ3**. We compare MERC Flash, MERC, Trix and Flock using pre-training overhead and zero-shot inference profiling. The measurements include parameter count, training and inference time per batch, peak GPU memory, peak CPU memory and zero-shot performance. Bounded SPD computation in MERC is included in the reported training and inference latency. Inference stress tests are conducted on unseen graphs of increasing scale and density, including the HM inductive series (Hamaguchi et al., 2017), Hetionet (Himmelstein et al., 2017) and YAGO310 (Mahdisoltani et al., 2015). All models are profiled under a unified hardware and batch-size setting whenever feasible (a single NVIDIA H100 80GB GPU, batch size 8). When a model fails due to memory limits, the failure is reported rather than excluded. For Flock, official evaluation settings are used when feasible, while memory-driven walk-number reductions are applied only to datasets that otherwise trigger OOM. he affected cases and exact evaluation profiles are reported in the Appendix I.

Results. Tables 4 separates the MERC family into two operating points. MERC Flash is the efficiency-oriented variant: it uses a single `Complex` branch and is designed to reduce inference cost while preserving much of the benefit of geometry-aware message passing. Full MERC is the accuracy- and robustness-oriented variant: it increases arithmetic cost through parallel geometric branches, but provides stronger average performance on complex or shifted graphs.

The scalability profile in Figure 4 further highlights this distinction. Flock provides a useful expressiveness reference, but its random-walk and sequence processing pipeline becomes increasingly expensive on dense or

Model	Parameter Count	Pre-Training Phase			Zero-Shot Inference Phase						
		Time / Batch	Peak VRAM	Peak RAM	Hetionet			YAGO310			
		(s)	(MB)	(MB)	Time (s)	VRAM (MB)	MRR	Time (s)	VRAM (MB)	MRR	Hits@10
Flock	801,969	0.796	26670.212	2098.613	5.808	26415.284	0.246		OOM		
Trix	87,138	0.593	4341.897	2388.888	0.450	1696.936	**0.279**	0.929	**1995.063**	0.409	0.627
MERC Flash	81,188	**0.085**	**3595.356**	**1141.559**	**0.111**	**1274.417**	0.236	**0.373**	2491.129	0.444	0.625
MERC	176,548	0.188	6916.390	1143.051	0.269	3338.611	**0.279**	1.059	8162.072	**0.510**	**0.684**

Table 4: Computation efficiency and zero-shot performance profiles. Pre-training metrics are profiled on the base graph, while zero-shot inference metrics track execution overhead and reasoning accuracy under a unified batch size of 8. The best results are highlighted in **bold**, the second best are underlined.

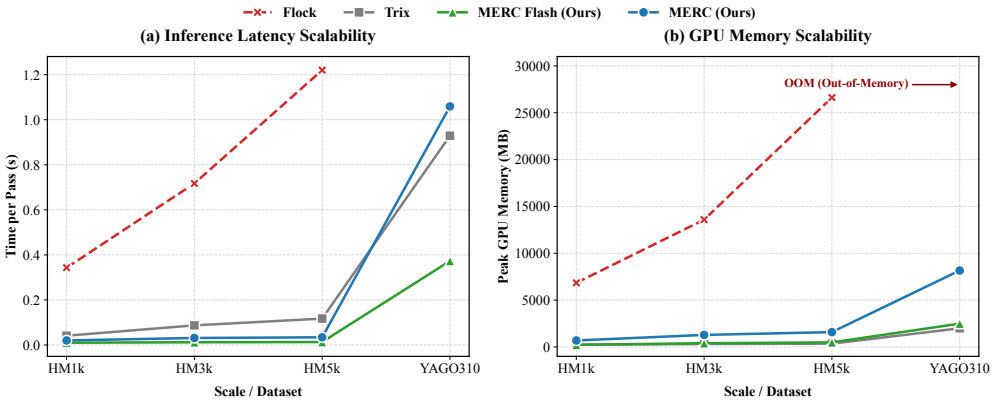


Figure 4: Scalability profiling of zero-shot inference across varying graph sizes. **(a)** Inference latency per forward pass. **(b)** Peak GPU memory allocation. The dashed red line illustrates Flock's rapid computational growth, which culminates in an Out-of-Memory (OOM) crash on the large-scale `YAGO310` dataset. In contrast, the MERC family maintains stable inference profiles under the tested graph scales.

large graphs and reaches memory limits under the largest stress setting. Trix is more stable in memory but suffers from slower temporal scaling as graph density and relation interactions increase. In contrast, MERC Flash defines the efficiency-oriented frontier, while MERC defines the accuracy-oriented frontier.

Takeaway (RQ3)

MERC provides two practical operating points for scalable zero-shot inference: MERC Flash offers a lightweight deterministic variant, while full MERC prioritizes accuracy on larger and more complex graphs. Compared with stochastic walk-based reasoning, MERC avoids severe sampling overhead and memory failures. Compared with dense updating, it offers a flexible balance between algebraic expressiveness and computational scalability.

5.5 Component Analysis and Ablation Studies

Impact of architectural components. This ablation evaluates the contribution of the main components introduced in Section 4: geometric message operators, structured relative position encoding, topology-aware relation refinement, layer-wise relation recalibration from entity context and query-guided branch fusion.

Table 5 shows that MERC's performance does not arise from a single isolated modification. The largest degradation occurs when layer-wise relation recalibration from entity context is removed. It indicates that MERC's gains are not solely due to adding geometric branches. Instead, geometry-aware propagation is most effective when relation states are dynamically updated with evolving entity-level evidence.

Model	Architectural Variation	Inductive e, r (23 graphs)		Inductive e (18 graphs)		Transductive (13 graphs)		Total Avg (54 graphs)	
		MRR	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**
$\text{MERC}_{\text{ablations}}$	w/o Multi-branch (DistMult)	0.362	0.537	0.437	0.592	0.338	0.493	0.381	0.545
	w/o PE	0.376	0.552	0.441	0.578	**0.354**	0.511	0.392	0.551
	w/o Layer-wise Co-evolution	0.353	0.535	0.437	0.588	0.333	0.492	0.376	0.542
	w/o Topology Attention	0.376	0.546	**0.446**	**0.594**	0.352	**0.512**	0.393	0.554
	w/o Query-Guided Attention	0.371	0.551	0.450	**0.599**	0.348	0.507	0.391	**0.557**
MERC Flash	w/o Multi-branch (Complex)	0.379	0.546	0.444	0.591	0.339	0.495	0.391	0.549
MERC	Full Model	**0.391**	**0.556**	0.446	0.588	0.347	0.509	**0.398**	**0.555**

Table 5: Average zero-shot link prediction results for the ablations of the main architectural components across 54 KGs. The best results among all architectural variations are highlighted in **bold**.

Replacing the multi-branch design with a single `DistMult`-style operator also weakens zero-shot transfer, which supports the importance of algebraic operator choice. Removing structured relative position encoding, topology-aware relation refinement or query-guided attention produces smaller but informative changes, suggesting that these modules act as supporting calibration mechanisms rather than the sole source of improvement. Overall, the ablations support the main claim that MERC is a coupled co-evolutionary multi-geometry framework.

Branch combinations. This ablation examines whether the selected geometric branches provide complementary information.

Model	Branch Combination	Inductive e, r (23 graphs)		Inductive e (18 graphs)		Transductive (13 graphs)		Total Avg (54 graphs)	
		MRR	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**
$\text{MERC}_{\text{combos}}$	Complex + Split-Complex	0.379	0.551	**0.451**	**0.597**	0.348	0.501	0.395	**0.555**
	Complex + DistMult	0.370	0.545	0.436	0.591	0.338	0.491	0.384	0.547
	Complex + Dual	0.378	0.548	0.439	0.588	**0.351**	**0.512**	0.392	0.553
	Complex + Split-Complex + DistMult	0.371	0.550	0.448	0.595	0.347	0.507	0.391	**0.555**
MERC	Complex + Split-Complex + Dual	**0.391**	**0.556**	0.446	0.588	0.347	0.509	**0.398**	**0.555**

Table 6: Average zero-shot link prediction results for the exploration of message function branch combinations across 54 KGs. The best results among all branch combinations are highlighted in **bold**.

Table 6 shows that the strongest configuration is not obtained by simply adding more operators, but by combining geometrically complementary ones. The full `Complex + Split-Complex + Dual` setting provides the best overall MRR and competitive Hits@10, suggesting a favorable balance across graph regimes. Adding a real-valued `DistMult` branch does not provide the same benefit. By contrast, the `Dual` branch contributes a non-redundant transformation, supporting MERC's design principle of using branches with distinct algebraic biases. Interestingly, some two-branch configurations are competitive in specific settings, this suggesting that the best branch choice may depend on the target graph regime.

Fusion strategies. This ablation compares how branch representations should be combined before scoring. It evaluates whether branch-specific representations should be preserved until final integration or collapsed early through simple summation.

Table 7 shows that preserving branch identity before final scoring is important for MERC. Concatenation-based fusion outperforms element-wise summation because it keeps the outputs of different geometric spaces separable until the model learns how to combine them. This result supports the query-guided fusion design: MERC should not merely aggregate geometric branches, but should retain their distinct relational signals before final integration.

Model	Branch Combination	Fusion Mechanism	Inductive e, r (23 graphs)		Inductive e (18 graphs)		Transductive (13 graphs)		Total Avg (54 graphs)	
			MRR	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**	**MRR**	**H@10**
$\text{MERC}_{\text{variants}}$	Complex + Split-Complex	sum	0.369	0.546	0.450	**0.598**	0.340	**0.501**	0.389	0.552
	Complex + Split-Complex	concat	**0.379**	**0.551**	**0.451**	0.597	**0.348**	**0.501**	**0.395**	**0.555**
	Complex + Dual	sum	0.369	0.546	0.435	**0.591**	0.340	0.495	0.384	0.549
	Complex + Dual	concat	**0.378**	**0.548**	**0.439**	0.588	**0.351**	**0.512**	**0.392**	**0.553**

Table 7: Average zero-shot link prediction results evaluating different fusion strategies across 54 KGs. The best results among each branch combination are highlighted in **bold**.

Takeaway (RQ4)

The ablation studies indicate that MERC's gains come from the interaction between expressive geometric operators and adaptive relation recalibration. Removing layer-wise relation recalibration causes the largest degradation, showing that multi-geometry propagation alone is not the whole explanation. Structural position encoding, topology-aware relation refinement and query-guided fusion provide supporting calibration, while branch-combination and fusion results suggest that preserving branch-specific geometric evidence is useful for zero-shot transfer.

6 Conclusion

We introduced MERC, a co-evolutionary multi-geometry structural KG reasoning model for zero-shot knowledge graph reasoning. MERC couples parallel Complex, Split-Complex and Dual message passing branches with layer-wise entity-relation recalibration, allowing complementary geometric entity propagation and adaptive relation states to inform each other during inference. Topology-aware relation refinement and query-guided fusion further support this process by refining marginalized relation graph representations and combining branch-specific evidence for each query.

We evaluated MERC on 54 zero-shot KG benchmarks and 31 biomedical graphs. The results show that MERC improves overall average performance over deterministic structural baselines and remains competitive with stochastic random-walk-based models, with particularly clear gains under unseen relation vocabularies, anti-symmetric and compositional relational patterns, and cross-domain biomedical shifts. The MetaFam case study, NELL-995 relation-embedding analysis and ablation studies further suggest that MERC's gains arise from the interaction between complementary geometric operators, dynamic calibration of entity and relation states and query-adaptive branch fusion.

MERC also has limitations. It does not uniformly dominate all baselines: Trix remains marginally stronger in highly homologous Inductive e settings, suggesting that fine-grained entity-indexed relation interaction can be especially useful when relation vocabularies are shared. MERC also increases arithmetic cost through parallel geometric branches, although MERC Flash provides an efficiency-oriented variant. Future work may develop adaptive branch selection or conditional computation mechanisms that activate only the most relevant geometric spaces for each query, reducing the cost of full MERC while preserving its expressiveness. Another direction is to extend co-evolutionary multi-geometry reasoning to temporal, hyper-relational and higher-order knowledge graphs, where relation semantics may depend on time, context or multi-entity interactions.

Broader Impact Statement

This work develops MERC, a structural knowledge graph reasoning model for zero-shot reasoning over unseen knowledge graphs. Its potential benefit is to improve transferable graph-based reasoning in domains where retraining or graph-specific annotation is costly, including ontology completion, scientific hypothesis generation and biomedical graph analysis.

The main risks concern misuse, overinterpretation and propagation of graph artifacts. Knowledge graph completion models predict plausible missing links rather than verified facts. In sensitive domains such as biomedicine, MERC's outputs should be treated only as hypotheses for further investigation, not as clinically validated discoveries, diagnostic recommendations or treatment guidance. Any downstream biomedical use should require independent expert review, domain-specific validation and appropriate safety checks before influencing scientific, clinical or policy decisions.

MERC may also inherit biases, incompleteness, outdated information or weakly supported relations from the knowledge graphs used for training, evaluation or deployment. If an input graph contains biased, erroneous or unrepresentative relations, the model may reproduce or amplify these artifacts through link prediction. This risk is especially important when predicted links concern people, diseases, institutions or other sensitive entities. Users should therefore inspect data provenance, document graph construction procedures and treat model outputs as candidates for verification rather than authoritative conclusions.

There is also a potential privacy and misuse risk if MERC is applied to private, proprietary or human-derived graphs. In such settings, predicted links could reveal or suggest sensitive associations that were not explicitly disclosed in the original data. Future applications should therefore consider consent, data licensing, access control, de-identification, auditing and compliance with applicable privacy and data-protection regulations before deployment.

Finally, although MERC avoids stochastic random-walk sampling, the full multi-branch variant increases arithmetic cost through parallel geometric message passing. Large-scale deployment should consider computational and energy costs. MERC Flash provides a more efficiency-oriented variant, but users should still evaluate the accuracy-efficiency trade-off for their intended application.

Author Contributions

If you'd like to, you may include a section for author contributions as is done in many journals. This is optional and at the discretion of the authors. Only add this information once your submission is accepted and deanonymized.

Acknowledgments

Use unnumbered third level headings for the acknowledgments. All acknowledgments, including those to funding agencies, go at the end of the paper. Only add this information once your submission is accepted and deanonymized.

References

Ralph Abboud, İsmail İlkan Ceylan, Thomas Lukasiewicz, and Tommaso Salvatori. Boxe: A box embedding model for knowledge base completion. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6dbbe6abe5f14af882ff977fc3f35501-Abstract.html.

Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Mikhail Galkin, Sahand Sharifzadeh, Asja Fischer, Volker Tresp, and Jens Lehmann. Bringing light into the dark: A large-scale evaluation of knowledge graph embedding models under a unified framework. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(12):8825–8845, 2022. doi: 10.1109/TPAMI.2021.3124805. URL https://doi.org/10.1109/TPAMI.2021.3124805.

Mais G Ammari, Cathy R Gresham, Fiona M McCarthy, and Bindu Nanduri. Hpidb 2.0: a curated database for host–pathogen interactions. *Database*, 2016:baw103, 2016.

Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. Multi-relational poincaré graph embeddings. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox,

and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 4465–4475, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/f8b932c70d0b2e6bf071729a4fa68dfc-Abstract.html.

Janos X Binder, Sune Pletscher-Frankild, Kalliopi Tsafou, Christian Stolte, Seán I O'Donoghue, Reinhard Schneider, and Lars Juhl Jensen. Compartments: unification and visualization of protein subcellular localization evidence. *Database*, 2014, 2014.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf.

Karin Breuer, Amir K Foroushani, Matthew R Laird, Carol Chen, Anastasia Sribnaia, Raymond Lo, Geoffrey L Winsor, Robert EW Hancock, Fiona SL Brinkman, and David J Lynn. Innatedb: systems biology of innate immunity and beyond—recent updates and continuing curation. *Nucleic acids research*, 41(D1): D1228–D1233, 2013.

Kevin R Brown and Igor Jurisica. Online predicted human interaction database. *Bioinformatics*, 21(9): 2076–2082, 2005.

Sven Buchholz and Gerald Sommer. On clifford neurons and clifford multi-layer perceptrons. *Neural Networks*, 21(7):925–935, 2008.

Annalisa Buniello, Jacqueline A L MacArthur, Maria Cerezo, Laura W Harris, James Hayhurst, Cinzia Malangone, Aoife McMahon, Joannella Morales, Edward Mountjoy, Elliot Sollis, et al. The nhgri-ebi gwas catalog of published genome-wide association studies, targeted arrays and summary statistics 2019. *Nucleic acids research*, 47(D1):D1005–D1012, 2019.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. Dual quaternion knowledge graph embeddings. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pp. 6894–6902. AAAI Press, 2021. doi: 10.1609/AAAI.V35I8.16850. URL https://doi.org/10.1609/aaai.v35i8.16850.

Debyani Chakravarty, Jianjiong Gao, Sarah Phillips, Ritika Kundra, Hongxin Zhang, Jiaojiao Wang, Julia E Rudolph, Rona Yaeger, Tara Soumerai, Moriah H Nissan, et al. Oncokb: a precision oncology knowledge base. *JCO precision oncology*, 1:1–16, 2017.

Ines Chami, Adva Wolf, Da-Cheng Juan, Frederic Sala, Sujith Ravi, and Christopher Ré. Low-dimensional hyperbolic knowledge graph embeddings. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pp. 6901–6914. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.ACL-MAIN.617. URL https://doi.org/10.18653/v1/2020.acl-main.617.

Emilie Chautard, Lionel Ballut, Nicolas Thierry-Mieg, and Sylvie Ricard-Blum. Matrixdb, a database focused on extracellular protein–protein and protein–carbohydrate interactions. *Bioinformatics*, 25(5): 690–691, 2009.

Yihong Chen, Pasquale Minervini, Sebastian Riedel, and Pontus Stenetorp. Relation prediction as an auxiliary training objective for improving multi-relational graph representations. In Danqi Chen, Jonathan Berant, Andrew McCallum, and Sameer Singh (eds.), *3rd Conference on Automated Knowledge Base Construction, AKBC 2021, Virtual, October 4-8, 2021*, 2021. doi: 10.24432/C54K5W. URL https://doi.org/10.24432/C54K5W.

Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Velickovic. Principal neighbourhood aggregation for graph nets. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/99cad265a1768cc2dd013f0e740300ae-Abstract.html.

Kelsy C Cotto, Alex H Wagner, Yang-Yang Feng, Susanna Kiwala, Adam C Coffman, Gregory Spies, Alex Wollam, Nicholas C Spies, Obi L Griffith, and Malachi Griffith. Dgidb 3.0: a redesign and expansion of the drug–gene interaction database. *Nucleic acids research*, 46(D1):D1068–D1073, 2018.

Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Roy McMorran, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. The comparative toxicogenomics database: update 2019. *Nucleic acids research*, 47(D1):D948–D954, 2019.

N Del-Toro, M Duesbury, M Koch, L Perfetto, A Shrivastava, D Ochoa, O Wagih, J Piñero, M Kotlyar, et al. Capturing variation impact on molecular interactions in the imex consortium mutations data set. *Nature communications*, 10(1):10, 2019.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In Sheila A. McIlraith and Kilian Q. Weinberger (eds.), *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 1811–1818. AAAI Press, 2018. doi: 10.1609/AAAI.V32I1.11573. URL https://doi.org/10.1609/aaai.v32i1.11573.

Boyang Ding, Quan Wang, Bin Wang, and Li Guo. Improving knowledge graph embedding using simple constraints. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 110–121, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1011. URL https://aclanthology.org/P18-1011/.

Enjun Du, Siyi Liu, and Yongqi Zhang. Graphoracle: A foundation model for knowledge graph reasoning. *CoRR*, abs/2505.11125, 2025. doi: 10.48550/ARXIV.2505.11125. URL https://doi.org/10.48550/arXiv.2505.11125.

Sara El-Gebali, Jaina Mistry, Alex Bateman, Sean R Eddy, Aurélien Luciani, Simon C Potter, Matloob Qureshi, Lorna J Richardson, Gustavo A Salazar, Alfredo Smart, et al. The pfam protein families database in 2019. *Nucleic acids research*, 47(D1):D427–D432, 2019.

Antonio Fabregat, Steven Jupe, Lisa Matthews, Konstantinos Sidiropoulos, Marc Gillespie, Phani Garapati, Robin Haw, Bijay Jassal, Florian Korninger, Bruce May, et al. The reactome pathway knowledgebase. *Nucleic acids research*, 46(D1):D649–D655, 2018.

Mikhail Galkin, Max Berrendorf, and Charles Tapley Hoyt. An open challenge for inductive link prediction on knowledge graphs. *CoRR*, abs/2203.01520, 2022. doi: 10.48550/ARXIV.2203.01520. URL https://doi.org/10.48550/arXiv.2203.01520.

Mikhail Galkin, Xinyu Yuan, Hesham Mostafa, Jian Tang, and Zhaocheng Zhu. Towards foundation models for knowledge graph reasoning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=jVEoydFOl9.

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 *39th IEEE International Conference on Data Engineering, ICDE 2023, Anaheim, CA, USA, April 3-7, 2023*, pp. 1221–1233. IEEE, 2023. doi: 10.1109/ICDE55515.2023.00098. URL https://doi.org/10.1109/ICDE55515.2023.00098.

Madalina Giurgiu, Julian Reinhard, Barbara Brauner, Irmtraud Dunger-Kaltenbach, Gisela Fobo, Goar Frishman, Corinna Montrone, and Andreas Ruepp. Corum: the comprehensive resource of mammalian protein complexes—2019. *Nucleic acids research*, 47(D1):D559–D563, 2019.

Alba Gutiérrez-Sacristán, Solene Grosdidier, Olga Valverde, Marta Torrens, Àlex Bravo, Janet Pinero, Ferran Sanz, and Laura I Furlong. Psygenet: a knowledge platform on psychiatric disorders and their genes. *Bioinformatics*, 31(18):3075–3077, 2015.

Takuo Hamaguchi, Hidekazu Oiwa, Masashi Shimbo, and Yuji Matsumoto. Knowledge transfer for out-of-knowledge-base entities : A graph neural network approach. In Carles Sierra (ed.), *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pp. 1802–1808. ijcai.org, 2017. doi: 10.24963/IJCAI.2017/250. URL https://doi.org/10.24963/ijcai.2017/250.

Daniel Scott 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, sep 2017. ISSN 2050-084X. doi: 10.7554/eLife.26726. URL https://doi.org/10.7554/eLife.26726.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia D'amato, Gerard De Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. Knowledge graphs. *ACM Comput. Surv.*, 54(4), July 2021a. ISSN 0360-0300. doi: 10.1145/3447772. URL https://doi.org/10.1145/3447772.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard De Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. Knowledge graphs. *ACM Computing Surveys (Csur)*, 54(4):1–37, 2021b.

Xingyue Huang, Pablo Barceló, Michael M. Bronstein, İsmail İlkan Ceylan, Mikhail Galkin, Juan L. Reutter, and Miguel Romero. How expressive are knowledge graph foundation models? *CoRR*, abs/2502.13339, 2025. doi: 10.48550/ARXIV.2502.13339. URL https://doi.org/10.48550/arXiv.2502.13339.

Timothy Jewison, Yilu Su, Fatemeh Miri Disfany, Yongjie Liang, Craig Knox, Adam Maciejewski, Jenna Poelzer, Jessica Huynh, You Zhou, David Arndt, et al. Smpdb 2.0: big improvements to the small molecule pathway database. *Nucleic acids research*, 42(D1):D478–D484, 2014.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pp. 687–696. The Association for Computer Linguistics, 2015. doi: 10.3115/V1/P15-1067. URL https://doi.org/10.3115/v1/p15-1067.

Jinwoo Kim, Xingyue Huang, Krzysztof Olejniczak, Kyungbin Min, Michael M. Bronstein, Seunghoon Hong, and İsmail İlkan Ceylan. Flock: A knowledge graph foundation model via learning on random walks. *CoRR*, abs/2510.01510, 2025. doi: 10.48550/ARXIV.2510.01510. URL https://doi.org/10.48550/arXiv.2510.01510.

Jaejun Lee, Chanyoung Chung, and Joyce Jiyoung Whang. Ingram: Inductive knowledge graph embedding via relation graphs. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 18796–18809. PMLR, 2023. URL https://proceedings.mlr.press/v202/lee23c.html.

Ke Liang, Lingyuan Meng, Meng Liu, Yue Liu, Wenxuan Tu, Siwei Wang, Sihang Zhou, Xinwang Liu, Fuchun Sun, and Kunlun He. A survey of knowledge graph reasoning on graph types: Static, dynamic, and multi-modal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9456–9478, 2024.

Luana Licata, Prisca Lo Surdo, Marta Iannuccelli, Alessandro Palma, Elisa Micarelli, Livia Perfetto, Daniele Peluso, Alberto Calderone, Luisa Castagnoli, and Gianni Cesareni. Signor 2.0, the signaling network open resource 2.0: 2019 update. *Nucleic acids research*, 48(D1):D504–D510, 2020.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In Blai Bonet and Sven Koenig (eds.), *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pp. 2181–2187. AAAI Press, 2015. doi: 10.1609/AAAI.V29I1.9491. URL https://doi.org/10.1609/aaai.v29i1.9491.

Jiawei Liu, Cheng Yang, Zhiyuan Lu, Junze Chen, Yibo Li, Mengmei Zhang, Ting Bai, Yuan Fang, Lichao Sun, Philip S Yu, et al. Graph foundation models: Concepts, opportunities and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Shuwen Liu, Bernardo Grau, Ian Horrocks, and Egor Kostylev. Indigo: Gnn-based inductive knowledge graph completion using pair-wise encoding. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 2034–2045. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/0fd600c953cde8121262e322ef09f70e-Paper.pdf.

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 Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 5694–5703, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.459. URL https://aclanthology.org/2020.emnlp-main.459/.

Farzaneh Mahdisoltani, Joanna Asia Biega, and Fabian M. Suchanek. Yago3: A knowledge base from multilingual wikipedias. In *Conference on Innovative Data Systems Research*, 2015. URL https://api.semanticscholar.org/CorpusID:6611164.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. Commonsense knowledge base completion with structural and semantic context. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 2925–2933. AAAI Press, 2020. doi: 10.1609/AAAI.V34I03.5684. URL https://doi.org/10.1609/aaai.v34i03.5684.

Antonio Rueda Martin, Eleanor Williams, Rebecca E Foulger, Sarah Leigh, Louise C Daugherty, Olivia Niblock, Ivone US Leong, Katherine R Smith, Oleg Gerasimenko, Eik Haraldsdottir, et al. Panelapp crowdsources expert knowledge to establish consensus diagnostic gene panels. *Nature genetics*, 51(11): 1560–1565, 2019.

Mojtaba Nayyeri, Sahar Vahdati, Can Aykul, and Jens Lehmann. 5* knowledge graph embeddings with projective transformations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35(10), pp. 9064–9072, 2021a.

Mojtaba Nayyeri, Chengjin Xu, Mirza Mohtashim Alam, Jens Lehmann, and Hamed Shariat Yazdi. Logicenn: A neural based knowledge graphs embedding model with logical rules. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6):7050–7062, 2021b.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In Lise Getoor and Tobias Scheffer (eds.), *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pp. 809–816. Omnipress, 2011. URL https://icml.cc/2011/papers/438_icmlpaper.pdf.

Sandra Orchard, Mais Ammari, Bruno Aranda, Lionel Breuza, Leonardo Briganti, Fiona Broackes-Carter, Nancy H Campbell, Gayatri Chavali, Carol Chen, Noemi Del-Toro, et al. The mintact project—intact as a common curation platform for 11 molecular interaction databases. *Nucleic acids research*, 42(D1): D358–D363, 2014.

Oana Palasca, Alberto Santos, Christian Stolte, Jan Gorodkin, and Lars Juhl Jensen. Tissues 2.0: an integrative web resource on mammalian tissue expression. *Database*, 2018:bay003, 2018.

Sonia Pavan, Kathrin Rommel, María Elena Mateo Marquina, Sophie Höhn, Valérie Lanneau, and Ana Rath. Clinical practice guidelines for rare diseases: the orphanet database. *PloS one*, 12(1):e0170365, 2017.

Janet Piñero, Núria Queralt-Rosinach, Alex Bravo, Jordi Deu-Pons, Anna Bauer-Mehren, Martin Baron, Ferran Sanz, and Laura I Furlong. Disgenet: a discovery platform for the dynamical exploration of human diseases and their genes. *Database*, 2015:bav028, 2015.

Sune Pletscher-Frankild, Albert Pallejà, Kalliopi Tsafou, Janos X Binder, and Lars Juhl Jensen. Diseases: Text mining and data integration of disease–gene associations. *Methods*, 74:83–89, 2015.

Fredrik Pontén, Karin Jirström, and Matthias Uhlen. The human protein atlas—a tool for pathology. *The Journal of Pathology: A Journal of the Pathological Society of Great Britain and Ireland*, 216(4):387–393, 2008.

Tara Safavi and Danai Koutra. CoDEx: A Comprehensive Knowledge Graph Completion Benchmark. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 8328–8350, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.669. URL `https://aclanthology.org/2020.emnlp-main.669/`.

Lukasz Salwinski, Christopher S Miller, Adam J Smith, Frank K Pettit, James U Bowie, and David Eisenberg. The database of interacting proteins: 2004 update. *Nucleic acids research*, 32(suppl_1):D449–D451, 2004.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL `https://openreview.net/forum?id=HkgEQnRqYQ`.

Damian Szklarczyk, Alberto Santos, Christian Von Mering, Lars Juhl Jensen, Peer Bork, and Michael Kuhn. Stitch 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic acids research*, 44(D1):D380–D384, 2016.

Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. String v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1):D607–D613, 2019.

David Tamborero, Carlota Rubio-Perez, Jordi Deu-Pons, Michael P Schroeder, Ana Vivancos, Ana Rovira, Ignasi Tusquets, Joan Albanell, Jordi Rodon, Josep Tabernero, et al. Cancer genome interpreter annotates the biological and clinical relevance of tumor alterations. *Genome medicine*, 10(1):25, 2018.

Komal K. Teru, Etienne G. Denis, and William L. Hamilton. Inductive relation prediction by subgraph reasoning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9448–9457. PMLR, 2020. URL `http://proceedings.mlr.press/v119/teru20a.html`.

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, 2015. doi: 10.18653/v1/W15-4007. URL `https://aclanthology.org/W15-4007`.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In Maria-Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 2071–2080. JMLR.org, 2016. URL `http://proceedings.mlr.press/v48/trouillon16.html`.

The UniProt Consortium. Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 46(5): 2699–2699, 2018.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In Carla E. Brodley and Peter Stone (eds.), *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, pp. 1112–1119. AAAI Press, 2014. doi: 10.1609/AAAI.V28I1.8870. URL https://doi.org/10.1609/aaai.v28i1.8870.

Frank Wilcoxon. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83, 1945. ISSN 00994987. URL http://www.jstor.org/stable/3001968.

David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. Drugbank 5.0: a major update to the drugbank database for 2018. *Nucleic acids research*, 46(D1):D1074–D1082, 2018.

Wenhan Xiong, Thien Hoang, and William Yang Wang. DeepPath: A reinforcement learning method for knowledge graph reasoning. In Martha Palmer, Rebecca Hwa, and Sebastian Riedel (eds.), *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 564–573, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1060. URL https://aclanthology.org/D17-1060/.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6575.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 28877–28888, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f1c1592588411002af340cbaedd6fc33-Abstract.html.

Andreas Zanzoni, Luisa Montecchi-Palazzi, Michele Quondam, Gabriele Ausiello, Manuela Helmer-Citterich, and Gianni Cesareni. Mint: a molecular interaction database. *FEBS letters*, 513(1):135–140, 2002.

Muhan Zhang, Pan Li, Yinglong Xia, Kai Wang, and Long Jin. Labeling trick: A theory of using graph neural networks for multi-node representation learning. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 9061–9073, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/4be49c79f233b4f4070794825c323733-Abstract.html.

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. Quaternion knowledge graph embeddings. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 2731–2741, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/d961e9f236177d65d21100592edb0769-Abstract.html.

Yongqi Zhang and Quanming Yao. Knowledge graph reasoning with relational digraph. In Frédérique Laforest, Raphaël Troncy, Elena Simperl, Deepak Agarwal, Aristides Gionis, Ivan Herman, and Lionel Médini (eds.), *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, pp. 912–924. ACM, 2022. doi: 10.1145/3485447.3512008. URL https://doi.org/10.1145/3485447.3512008.

Yongqi Zhang, Zhanke Zhou, Quanming Yao, Xiaowen Chu, and Bo Han. Adaprop: Learning adaptive propagation for graph neural network based knowledge graph reasoning. In Ambuj K. Singh, Yizhou Sun, Leman Akoglu, Dimitrios Gunopulos, Xifeng Yan, Ravi Kumar, Fatma Ozcan, and Jieping Ye (eds.), *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, pp. 3446–3457. ACM, 2023. doi: 10.1145/3580305.3599404. URL https://doi.org/10.1145/3580305.3599404.

Yucheng Zhang, Beatrice Bevilacqua, Mikhail Galkin, and Bruno Ribeiro. TRIX: A more expressive model for zero-shot domain transfer in knowledge graphs. *CoRR*, abs/2502.19512, 2025. doi: 10.48550/ARXIV.2502.19512. URL https://doi.org/10.48550/arXiv.2502.19512.

Jincheng Zhou, Beatrice Bevilacqua, and Bruno Ribeiro. An OOD multi-task perspective for link prediction with new relation types and nodes. *CoRR*, abs/2307.06046, 2023. doi: 10.48550/ARXIV.2307.06046. URL https://doi.org/10.48550/arXiv.2307.06046.

Jincheng Zhou, Yucheng Zhang, Jianfei Gao, Yangze Zhou, and Bruno Ribeiro. Double equivariance for inductive link prediction for both new nodes and new relation types, 2025. URL https://arxiv.org/abs/2302.01313.

Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal A. C. Xhonneux, and Jian Tang. Neural bellman-ford networks: A general graph neural network framework for link prediction. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 29476–29490, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/f6a673f09493afcd8b129a0bcf1cd5bc-Abstract.html.

Zhaocheng Zhu, Xinyu Yuan, Michael Galkin, Louis-Pascal A. C. Xhonneux, Ming Zhang, Maxime Gazeau, and Jian Tang. A*net: A scalable path-based reasoning approach for knowledge graphs. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/b9e98316cb72fee82cc1160da5810abc-Abstract-Conference.html.

A Complexity Analysis

This appendix provides the detailed complexity analysis of MERC under the actual model configuration used in our experiments. Unless otherwise specified, MERC uses L propagation layers, hidden dimension d, relation hidden dimension d_r, H attention heads and $M = |\mathcal{M}|$ entity-propagation branches. MERC-Flash uses the same depth and hidden dimension but sets $M = 1$. The entity aggregation function is `sum`. The final branch feature dimension used by the query-guided fusion module is d_f.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{R}, \mathcal{E})$ be the entity-level knowledge graph, where $|\mathcal{V}|$ is the number of entities, $|\mathcal{E}|$ is the number of directed entity edges after inverse edge augmentation and $|\mathcal{R}|$ is the number of directed relation types. Let $\mathcal{G}r = (\mathcal{R}, \mathcal{R}\text{fund}, \mathcal{E}_r)$ denote the marginalized relation graph used by MERC. Let B be the batch size and K the number of candidate tail entities scored per query. For sampled training, K is the number of positive and negative candidates in the batch, for full ranking, $K = |\mathcal{V}|$.

A.1 Size of the Marginalized Relation Graph

MERC constructs its relation graph differently from entity-indexed relation-adjacency methods. The input entity graph is assumed to already contain inverse edges. For each entity $v \in \mathcal{V}$, define the sets of relation types appearing at v in the head and tail roles as

$$\mathcal{R}_h(v) = \{r \in \mathcal{R} : \exists u,\ (v, r, u) \in \mathcal{E}\}, \qquad \mathcal{R}_t(v) = \{r \in \mathcal{R} : \exists u,\ (u, r, v) \in \mathcal{E}\}. \tag{33}$$

Let

$$a_h(v) = |\mathcal{R}_h(v)|, \qquad a_t(v) = |\mathcal{R}_t(v)|, \tag{34}$$

and define the maximum role-aware incident relation count as

$$\alpha_{\pm} = \max_{v \in \mathcal{V}} \big(a_h(v) + a_t(v)\big). \tag{35}$$

Equivalently, if one defines

$$\alpha = \max_{v \in \mathcal{V}} |\mathcal{R}_h(v) \cup \mathcal{R}_t(v)|, \tag{36}$$

then $\alpha_{\pm} \leq 2\alpha$.

MERC first deduplicates entity–relation incidences. Let

$$E_h = \{(v, r) : r \in \mathcal{R}_h(v)\}, \qquad E_t = \{(v, r) : r \in \mathcal{R}_t(v)\}. \tag{37}$$

The implementation represents E_h and E_t as sparse incidence matrices

$$\mathbf{E}_h, \mathbf{E}_t \in \{0, 1\}^{|\mathcal{V}| \times |\mathcal{R}|}. \tag{38}$$

It also constructs degree-normalized transposes $\mathbf{E}_h^\top$ and $\mathbf{E}_t^\top$, whose nonzero values are scaled by the corresponding entity-level incidence degrees. The four relation-pair topology matrices are then computed by sparse matrix multiplication:

$$\begin{aligned} \mathbf{A}_{hh} &= \mathbf{E}_h^\top \mathbf{E}_h, & \mathbf{A}_{tt} &= \mathbf{E}_t^\top \mathbf{E}_t, \\ \mathbf{A}_{ht} &= \mathbf{E}_h^\top \mathbf{E}_t, & \mathbf{A}_{th} &= \mathbf{E}_t^\top \mathbf{E}_h. \end{aligned} \tag{39}$$

Each matrix has shape $|\mathcal{R}| \times |\mathcal{R}|$. The final relation graph is obtained by taking the nonzero indices of these four matrices and assigning them one of four topology edge types: hh, tt, ht, th. Therefore, MERC's relation graph has $|\mathcal{R}|$ nodes and only 4 relation-edge types. The mediating entity is not used as an edge type and is not retained as an additional graph dimension.

We now bound the number of relation-graph edges. For a fixed entity v, the four products can induce at most

$$a_h(v)^2, \qquad a_t(v)^2, \qquad a_h(v)a_t(v), \qquad a_t(v)a_h(v) \tag{40}$$

ordered relation pairs for hh, tt, ht, and th, respectively. Thus, the total number of relation-pair incidences induced by v is at most

$$a_h(v)^2 + a_t(v)^2 + 2a_h(v)a_t(v) = \big(a_h(v) + a_t(v)\big)^2. \tag{41}$$

Summing over all entities gives

$$|\mathcal{E}_r| \leq \sum_{v \in \mathcal{V}} \big(a_h(v) + a_t(v)\big)^2 \leq |\mathcal{V}|\alpha_{\pm}^2. \tag{42}$$

This is an entity-incidence upper bound on the number of relation pairs that can be generated before duplicate relation pairs are merged.

However, MERC marginalizes over the mediating entity through the sparse products above. After coalescing, a typed relation edge (r_i, r_j, τ) is stored once if at least one entity induces it, regardless of how many entities induce the same typed pair. Since there are $|\mathcal{R}|^2$ ordered relation pairs and four topology types, we also have the global support bound

$$|\mathcal{E}_r| \leq 4|\mathcal{R}|^2. \tag{43}$$

Combining Eq. equation 42 and Eq. equation 43, the marginalized relation graph used by MERC satisfies

$$|\mathcal{E}_r| \leq \min\left\{|\mathcal{V}|\alpha_{\pm}^2,\ 4|\mathcal{R}|^2\right\}. \tag{44}$$

Using the union-based incident relation count α, this can equivalently be written as

$$|\mathcal{E}_r| \leq \min\left\{4|\mathcal{V}|\alpha^2,\ 4|\mathcal{R}|^2\right\}. \tag{45}$$

Comparison with entity-indexed relation adjacency. An entity-indexed construction stores relation interactions separately for each mediating entity, producing entries of the form (r_i, r_j, v, τ). Such a representation has an entity-indexed relation-adjacency tensor with an explicit $|\mathcal{V}|$ dimension, and its number of stored relation-pair entries can scale as

$$\mathcal{O}(|\mathcal{V}|\alpha_{\pm}^2). \tag{46}$$

In contrast, MERC stores only the marginalized typed relation-pair support (r_i, r_j, τ). The mediating entities contribute to the sparse matrix product values but are not retained as separate edge indices. Consequently, MERC's relation graph is bounded by

$$\mathcal{O}(|\mathcal{R}|^2) \tag{47}$$

in the worst case with respect to the relation vocabulary size, and the relation-state refinement step does not contain an unavoidable entity-indexed relation-pair materialization term.

A.2 Bounded Shortest-Path Position Encoding

MERC computes a bounded shortest-path distance from each query head entity to all entities. The implementation performs a parallel breadth-limited propagation for at most L steps. At each step, the active frontier is propagated along entity edges and reduced by a scatter operation. Therefore,

$$T_{\text{spd}} = \mathcal{O}(BL(|\mathcal{E}| + |\mathcal{V}|)) \,. \tag{48}$$

The distance embedding lookup and gated position injection add

$$T_{\text{pe}} = \mathcal{O}(B|\mathcal{V}|d + |\mathcal{E}| + Bd) \,, \tag{49}$$

where the $|\mathcal{E}|$ term comes from counting relation frequencies over entity edges. Since L and d are fixed, this stage is linear in $|\mathcal{E}| + |\mathcal{V}|$.

A.3 Multi-Branch Entity Graph Propagation

For each layer and each branch, MERC applies a generalized relational convolution over the entity graph. Under the actual configuration, the aggregation function is `sum`, we set $d = d_l$ in the complexity analysis. Thus, for one layer and one branch, the fused relational sparse matrix–matrix multiplication has cost

$$\mathcal{O}(B|\mathcal{E}|d) \,. \tag{50}$$

The boundary addition and node update cost

$$\mathcal{O}\big(B|\mathcal{V}|d + B|\mathcal{V}|d^2\big) = \mathcal{O}\big(B|\mathcal{V}|d^2\big) \,. \tag{51}$$

Because MERC uses projected relation states inside the entity propagation layer, the layer-specific relation projection adds

$$\mathcal{O}\big(B|\mathcal{R}|d^2\big) \,. \tag{52}$$

Therefore, across M branches and L layers,

$$T_{\text{entity}} = \mathcal{O}\big(MLB\left(|\mathcal{E}|d + |\mathcal{V}|d^2 + |\mathcal{R}|d^2\right)\big) \,. \tag{53}$$

With the actual fixed configuration L, d and M, this becomes

$$T_{\text{entity}} = \mathcal{O}(B(|\mathcal{E}| + |\mathcal{V}| + |\mathcal{R}|)) \,. \tag{54}$$

A.4 Global Entity Context Pooling

After each entity-propagation layer, MERC concatenates the M branch states and computes four global statistical context tokens:

$$\text{mean}, \quad \text{max}, \quad \text{min}, \quad \text{std}. \tag{55}$$

The concatenated entity representation has shape

$$B \times |\mathcal{V}| \times (Md). \tag{56}$$

Computing the four statistics over the entity dimension costs

$$T_{\text{pool}} = \mathcal{O}(LB|\mathcal{V}|Md)\,. \tag{57}$$

Since the number of statistics is fixed to S, the output context has shape

$$B \times S \times (Md), \tag{58}$$

which is independent of $|\mathcal{V}|$ after pooling.

A.5 Marginalized Relation-State Refinement

At each layer, MERC refines relation states on $\mathcal{G}_r$ using topology-aware relation convolution. The query, key, value and output projections over relation nodes cost

$$\mathcal{O}\big(B|\mathcal{R}|d_r^2\big)\,, \tag{59}$$

while topology-aware attention message passing over relation graph edges costs

$$\mathcal{O}(B|\mathcal{E}_r|d_r)\,. \tag{60}$$

Thus,

$$T_{\text{rel}} = \mathcal{O}\big(LB\left(|\mathcal{E}_r|d_r + |\mathcal{R}|d_r^2\right)\big)\,. \tag{61}$$

For fixed L and d_r, the worst-case relation-state refinement term is

$$\mathcal{O}\big(B|\mathcal{R}|^2\big)\,. \tag{62}$$

A.6 Entity-to-Relation Co-Evolution

The co-evolution module lets relation states attend to the four global entity context tokens, not to all entity nodes. The relation queries have shape

$$B \times |\mathcal{R}| \times d_r, \tag{63}$$

whereas the entity-side keys and values have shape

$$B \times S \times d_r. \tag{64}$$

The relation-side query and output projections cost

$$\mathcal{O}\big(B|\mathcal{R}|d_r^2\big)\,. \tag{65}$$

The key and value projections from entity context dimension Md to d_r cost

$$\mathcal{O}(BSMdd_r)\,. \tag{66}$$

The attention matrix has size

$$B \times H \times |\mathcal{R}| \times S, \tag{67}$$

so attention scoring and value aggregation cost

$$\mathcal{O}(BS|\mathcal{R}|d_r)\,. \tag{68}$$

Therefore,

$$T_{\text{coev}} = \mathcal{O}\big(LB\left(|\mathcal{R}|d_r^2 + SMdd_r + S|\mathcal{R}|d_r\right)\big)\,. \tag{69}$$

Since S, M, $d = d_r$, and L are fixed, this reduces to

$$T_{\text{coev}} = \mathcal{O}(B|\mathcal{R}|)\,. \tag{70}$$

This is substantially cheaper than dense entity–relation cross-attention, which would require

$$\mathcal{O}(B|\mathcal{V}||\mathcal{R}|d_r) \tag{71}$$

because every relation would attend to every entity.

A.7 Query-Guided Candidate Fusion

MERC scores K candidate tail entities per query. The query-guided branch attention and the final MLP operate on tensors of size

$$B \times K \times M \times d_f, \tag{72}$$

The query projection, key and value projections, branch attention, and final MLP together cost

$$T_{\text{fusion}} = \mathcal{O}\big(BKM^2 d_f^2\big)\,. \tag{73}$$

For MERC-Flash with $M = 1$, the branch attention is bypassed and the fusion cost becomes

$$T_{\text{fusion}}^{\text{Flash}} = \mathcal{O}\big(BKd_f^2\big)\,. \tag{74}$$

Since M and d_f are fixed constants, fusion is linear in K. In full ranking, $K = |\mathcal{V}|$, giving

$$T_{\text{fusion}} = \mathcal{O}(B|\mathcal{V}|)\,. \tag{75}$$

A.8 Overall Time Complexity

Combining the above components, the full forward-pass complexity of MERC is

$$\begin{aligned} T_{\text{MERC}} = \mathcal{O}\Big(& BL(|\mathcal{E}| + |\mathcal{V}|) + B|\mathcal{V}|d + |\mathcal{E}| \\ & + MLB\left(|\mathcal{E}|d + |\mathcal{V}|d^2 + |\mathcal{R}|d^2\right) \\ & + LB|\mathcal{V}|Md + LB\left(|\mathcal{E}_r|d_r + |\mathcal{R}|d_r^2\right) \\ & + LB\left(|\mathcal{R}|d_r^2 + SMdd_r + S|\mathcal{R}|d_r\right) + BKM^2 d_f^2\Big), \end{aligned} \tag{76}$$

Under the actual MERC configuration, L, $d = d_r = d_f$, S, and M for MERC or $M = 1$ for MERC-Flash are fixed architectural constants. Thus, Eq. equation 76 reduces to

$$T_{\text{MERC}} = \mathcal{O}(B\left(|\mathcal{E}| + |\mathcal{V}| + |\mathcal{E}_r| + |\mathcal{R}| + K\right) + |\mathcal{E}|)\,. \tag{77}$$

For fixed batch size and full-ranking evaluation, $K = |\mathcal{V}|$, giving

$$T_{\text{MERC}} = \mathcal{O}(|\mathcal{E}| + |\mathcal{V}| + |\mathcal{E}_r| + |\mathcal{R}|)\,. \tag{78}$$

By construction, MERC's relation graph is marginalized over mediating entities. Its edge count satisfies

$$|\mathcal{E}_r| \le \min\left\{|\mathcal{V}|\alpha_\pm^2, 4|\mathcal{R}|^2\right\}, \tag{79}$$

where $\alpha_\pm = \max_{v\in\mathcal{V}}\left(|\mathcal{R}_h(v)| + |\mathcal{R}_t(v)|\right)$ is the maximum role-aware incident relation count. Equivalently, if $\alpha = \max_v |\mathcal{R}_h(v) \cup \mathcal{R}_t(v)|$, then $\alpha_\pm \le 2\alpha$ and

$$|\mathcal{E}_r| \le \min\left\{4|\mathcal{V}|\alpha^2, 4|\mathcal{R}|^2\right\}. \tag{80}$$

Therefore, a more precise graph-size-dependent bound is

$$T_{\text{MERC}} = \mathcal{O}\big(|\mathcal{E}| + |\mathcal{V}| + \min\left\{|\mathcal{V}|\alpha_\pm^2, |\mathcal{R}|^2\right\}\big)\,. \tag{81}$$

Since

$$\min\left\{|\mathcal{V}|\alpha_\pm^2, |\mathcal{R}|^2\right\} \le |\mathcal{R}|^2, \tag{82}$$

the simplified upper bound reported in the main text follows:

$$T_{\text{MERC}} = \mathcal{O}\big(|\mathcal{E}| + |\mathcal{V}| + |\mathcal{R}|^2\big)\,. \tag{83}$$

This bound should be understood as the per-forward-pass complexity after the marginalized relation graph has been constructed. The relation graph is a static graph structure and can be precomputed once for a fixed knowledge graph. If relation-graph construction itself is included as preprocessing, its cost is governed by deduplicating entity–relation incidences and forming the four sparse products $\mathbf{E}_h^\top\mathbf{E}_h$, $\mathbf{E}_t^\top\mathbf{E}_t$, $\mathbf{E}_h^\top\mathbf{E}_t$, and $\mathbf{E}_t^\top\mathbf{E}_h$, the number of output nonzeros is precisely $|\mathcal{E}_r|$ and is bounded by Eq. equation 79.

A.9 Space and Memory Complexity

The main memory bottleneck in entity-level message passing is the materialization of edge-level message tensors. Without fused sparse propagation, an M-branch implementation would need to store tensors of shape

$$B \times |\mathcal{E}| \times d \tag{84}$$

for each layer and branch, leading to an edge-message activation memory term

$$\mathcal{O}(MLB|\mathcal{E}|d)\,. \tag{85}$$

MERC avoids this entity-edge materialization through fused relational sparse matrix–matrix multiplication in the entity propagation module. The message transformation and scatter-reduce aggregation are fused, so the dominant entity-side activation memory is node-bound:

$$\mathcal{O}(MLB|\mathcal{V}|d)\,. \tag{86}$$

The full persistent activation memory also includes relation states, relation-graph attention activations, candidate fusion states and bounded shortest-path states. A strict memory bound is therefore

$$\begin{aligned}\mathcal{S}_{\text{MERC}} = \mathcal{O}\Big(&MLB|\mathcal{V}|d + LB|\mathcal{R}|d_r + LB|\mathcal{E}_r|d_r \\ &+ BKMd_f + B|\mathcal{V}|\Big).\end{aligned} \tag{87}$$

The first term stores entity activations across branches and layers. The second term stores relation-node states. The third term accounts for topology-aware attention over the marginalized relation graph. The fourth term stores candidate branch features for query-guided fusion. The last term accounts for bounded shortest-path labels and frontier states.

Using the MERC relation-graph bound

$$|\mathcal{E}_r| \leq \min\left\{|\mathcal{V}|\alpha_{\pm}^2, 4|\mathcal{R}|^2\right\}, \tag{88}$$

Eq. equation 87 becomes

$$\begin{aligned}\mathcal{S}_{\text{MERC}} = \mathcal{O}\Big(&MLB|\mathcal{V}|d + LB|\mathcal{R}|d_r + LB\min\left\{|\mathcal{V}|\alpha_{\pm}^2, |\mathcal{R}|^2\right\}d_r \\ &+ BKMd_f + B|\mathcal{V}|\Big).\end{aligned} \tag{89}$$

Under the actual MERC configuration, L, M, d, d_r, and d_f are fixed constants. Thus,

$$\mathcal{S}_{\text{MERC}} = \mathcal{O}(B\left(|\mathcal{V}| + |\mathcal{R}| + |\mathcal{E}_r| + K\right))\,. \tag{90}$$

For full ranking, $K = |\mathcal{V}|$, and hence

$$\mathcal{S}_{\text{MERC}} = \mathcal{O}(B\left(|\mathcal{V}| + |\mathcal{R}| + |\mathcal{E}_r|\right))\,. \tag{91}$$

Using $|\mathcal{E}_r| \leq 4|\mathcal{R}|^2$, this gives the worst-case bound

$$\mathcal{S}_{\text{MERC}} = \mathcal{O}\big(B\left(|\mathcal{V}| + |\mathcal{R}|^2\right)\big)\,. \tag{92}$$

In standard knowledge graphs, $|\mathcal{R}| \ll |\mathcal{V}|$, and the relation graph is much smaller than the entity graph. Therefore, with fixed relation vocabulary and fused entity propagation, the dominant activation memory scales linearly with the number of entities:

$$\mathcal{S}_{\text{MERC}} = \mathcal{O}(|\mathcal{V}|) \quad \text{with respect to entity-graph size.} \tag{93}$$

Finally, the static storage of the input graphs is separate from differentiable activation memory. It requires

$$\mathcal{O}(|\mathcal{E}| + |\mathcal{E}_r|) \tag{94}$$

space for the entity graph and the marginalized relation graph. This static storage does not correspond to materialized per-layer entity-edge messages and is not multiplied by B, L, or M.

B Expressivity Analysis

Analysis: Expressivity of Multi-Space Geometric Reasoning

MERC exhibits expanded expressivity over Trix (Zhang et al., 2025) and Ultra (Galkin et al., 2024) in resolving topological symmetries and approximating complex relational compositions over paths.

Discussion. The entity level expressivity of MERC is enhanced through two mathematical mechanisms:

- **Topological Distinguishability via Absolute Geometry:** While Trix and Ultra employ conditional message passing with source node anchoring (elevating expressivity beyond standard 1-WL), they inherently rely on the unrolling of localized computation trees. In highly symmetric, locally isomorphic subgraphs, structural anchoring alone may struggle to differentiate nodes with identical local trees but different absolute distances from the query. MERC breaks this residual symmetry by computing the batched Shortest Path Distance (SPD) and injecting it via dynamic gating (PE). This explicit distance scalar provides an absolute geometric coordinate that aids in strict differentiability.

- **Algebraic Closure and Path Composition:** Trix and Ultra rely on the real-valued DistMult operator. While DistMult is inherently symmetric ($f(h, r, t) = f(t, r, h)$), Trix and Ultra mitigate this flaw along multi-hop paths by employing unshared, layer-specific relation weights ($W^{(l)}$). This step-wise parameterization breaks strict commutativity, allowing the model to memorize position dependent path sequences (i.e., $W_{r_1}^{(1)} \circ W_{r_2}^{(2)} \neq W_{r_2}^{(1)} \circ W_{r_1}^{(2)}$).

  However, this layer coupled memorization fundamentally lacks *algebraic closure* and suffers from poor cross-layer generalization. A true logical composition ($r_1 \circ r_2 \approx r_3$) should be structurally invariant regardless of the depth l at which it occurs. Trix and Ultra require re-learning these step dependent combinations for every layer.

  Conversely, MERC employs a complex-valued multi-space geometric ensemble. In the Complex space, relations are mapped as phase rotations in the complex plane $\mathbb{C}^{d_l}$. Path composition is resolved via Euler's formula as continuous phase additions:

$$\mathbf{h} \circ \mathbf{r}_1 \circ \mathbf{r}_2 = \mathbf{h} \circ e^{i(\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2)} \tag{95}$$

  This geometric transformation is inherently closed under composition and layer-agnostic. Furthermore, MERC's multi-space fusion spans multiple geometric manifolds: spherical (Complex), hyperbolic-like (Split-Complex) and translational (Dual). According to the Universal Approximation Theorem for geometric algebras Buchholz & Sommer (2008), this closed, multi-manifold basis subsumes the memorization-based representational capacity of a layer-coupled scalar operator, allowing MERC to approximate multiplex relation combinations more effectively.

C The MERC Algorithm Pipeline

To provide a comprehensive synthesis of the MERC architecture, this section formalizes the forward pass. The pipeline unifies the structural position encoding, the multi-branch entity evolution, the layer-wise relation co-evolution and the query-guided fusion.

Algorithm 1 Forward Pass of the MERC Architecture

Require: Knowledge graph $\mathcal{G}$, Marginalized relation graph $\mathcal{G}_r$
Require: Batch of queries $\mathcal{B} = \{(h_b, r_b, ?)\}_{b=1}^{B}$
Require: Message passing branches $\mathcal{M}$
Require: Model depth L
Ensure: Target entity prediction scores $\mathbf{Y} \in \mathbb{R}^{B\times N}$ with $\mathbf{Y}_{b,v} = \phi_b(v)$, Auxiliary entropy $\mathcal{H}(\boldsymbol{\alpha})$

1: // **1. Initialization and Structural PE**
2: Initialize relation representations $\mathbf{X}^{(0)} \in \mathbb{R}^{B\times R\times d}$ and extract query relations $\mathbf{q}_b = \mathbf{X}^{(0)}_{b,r_b}$
3: Initialize entity boundary features $\mathbf{H}^{(0)}_m \leftarrow \mathbf{0} \in \mathbb{R}^{B\times N\times d}$ for all $m \in \mathcal{M}$
4: Scatter query conditions to source nodes: $\mathbf{H}^{(0)}_{m,b,h_b} \leftarrow \mathbf{q}_b$ for all $m \in \mathcal{M}$
5: Compute SPD matrix $\mathbf{D} \in \mathbb{R}^{B\times N} \leftarrow \text{ParallelBFS}(\mathcal{B}, \mathcal{G}, d_{\max} = L + 1)$
6: Calculate dynamic PE gate g_{r_b} utilizing $\mathbf{q}_b$ and inference graph relation frequency f_{r_b}
7: Modulate positional embeddings: $\mathbf{P} \leftarrow \mathbf{E}_{\text{dist}}(\mathbf{D})$, $\tilde{\mathbf{P}}_{b,j} \leftarrow g_{r_b}\mathbf{P}_{b,j}$
8: Add structural prior to boundaries: $\mathbf{H}^{(0)}_m \leftarrow \mathbf{H}^{(0)}_m + \tilde{\mathbf{P}}$ for all $m \in \mathcal{M}$
9: // **2. Layer-wise Dual Graph Co-Evolution**
10: **for** $l = 1 \to L$ **do**
11: *// A. Parallel Entity Graph Evolution*
12: **for each** branch $m \in \mathcal{M}$ **in parallel do**
13: $\mathbf{H}^{(l)}_m \leftarrow \text{EntityUpdate}\left(\mathbf{H}^{(l-1)}_m, \mathbf{X}^{(l-1)}, \mu_m, \mathcal{G}\right)$ with residual connection and normalization
14: **end for**
15: *// B. Global Entity Context Pooling*
16: $\mathbf{C}^{(l)} \leftarrow \text{Concat}_{m\in\mathcal{M}}\left(\mathbf{H}^{(l)}_m\right)$
17: Compute aggregators $\mathbf{c}^{(l)}_{\text{mean}}, \mathbf{c}^{(l)}_{\max}, \mathbf{c}^{(l)}_{\min}, \mathbf{c}^{(l)}_{\text{std}}$ over the entity dimension N
18: $\mathbf{E}^{(l)}_{\text{ctx}} \leftarrow \text{Concat}_{\text{token}}\left(\mathbf{c}^{(l)}_{\text{mean}}, \mathbf{c}^{(l)}_{\max}, \mathbf{c}^{(l)}_{\min}, \mathbf{c}^{(l)}_{\text{std}}\right)$
19: *// C. Topology-Aware Relation Convolution*
20: $\tilde{\mathbf{X}}^{(l)} \leftarrow \text{TopologyAttentionConv}\left(\mathbf{X}^{(l-1)}, \mathcal{G}_r\right)$
21: *// D. Co-Evolution Cross-Attention*
22: $\mathbf{X}^{(l)} \leftarrow \text{CoEvolutionCrossAttention}(\tilde{\mathbf{X}}^{(l)}, \mathbf{E}^{(l)}_{\text{ctx}})$
23: **end for**
24: // **3. Query-Guided Fusion and Scoring**
25: Update batch query vectors utilizing the co-evolved relations: $\mathbf{q}^{\text{upd}}_b \leftarrow \mathbf{X}^{(L)}_{b,r_b}$
26: Extract terminal branch representations: $\mathbf{Z}_{\text{br}} \leftarrow \text{Stack}_{m\in\mathcal{M}}\left(\mathbf{H}^{(L)}_m\right)$
27: Construct branch representations $\mathbf{z}_{b,v,m}$ from terminal states and normalize them as $\bar{\mathbf{z}}_{b,v,m}$
28: $\mathbf{o}_{b,v}, \mathcal{H}(\boldsymbol{\alpha}) \leftarrow \text{QueryGuidedFusion}(\mathbf{q}^{\text{upd}}_b, \{\bar{\mathbf{z}}_{b,v,m}\}_{m\in\mathcal{M}})$
29: Compute scores for all N entities: $\mathbf{Y}_{b,v} \leftarrow \phi_b(v) = \text{MLP}_{\text{score}}([\mathbf{o}_{b,v} \parallel \mathbf{q}^{\text{upd}}_b])$
return $\mathbf{Y}, \mathcal{H}(\boldsymbol{\alpha})$

As depicted in Algorithm 1, the process is bound by $\mathcal{O}(L)$ sequential steps, corresponding to the outer layer loop. Within each iteration (Lines 12-14), message passing over the entity graph runs concurrently across the geometric spaces in $\mathcal{M}$. Dual graph synchronization is maintained by extracting the global entity state $\mathbf{E}^{(l)}_{\text{ctx}}$ (Line 19) and feeding it directly into the relation update mechanism (Lines 21-23).

D Implementation Details and Evaluation Protocol

This appendix provides the implementation details used for all zero-shot link prediction experiments, including the hardware and software environment, training hyperparameters, checkpoint-selection rule and statistical testing protocol.

Hardware and software environment. All pre-training stages were executed on an HPC cluster node equipped with four NVIDIA H200 GPUs, each with 141GB GPU memory. The software environment used Red Hat Enterprise Linux 9.4 (Plow), Python 3.9.25, PyTorch 2.1.0+cu118, PyTorch Geometric 2.4.0 and torch-scatter 2.1.2+pt21cu118 under CUDA 11.8. Zero-shot inference was performed on a single GPU whenever feasible. Depending on cluster allocation and memory requirements, inference used a single NVIDIA A100 40GB, H100 80GB or H200 141GB GPU.

Training hyperparameters. All MERC-family models were trained using the AdamW optimizer with a learning rate of 5.0×10^{-4}, batch size 16 and 10 training epochs. Each positive triple was paired with 512 negative samples, and adversarial negative sampling used a temperature of 1.0. A complete summary of the hyperparameters is provided in Table 8.

Hyperparameter	**Value**
Training Configurations	
Optimizer	AdamW
Learning Rate	5.0×10^{-4}
Batch Size	16
Number of Epochs	10
Batches per Epoch	80,000
Negative Samples	512
Adversarial Temperature	1.0
Strict Negative Sampling	True
Architecture Details (Entity & Relation Models)	
Hidden Dimensions	[32, 32, 32, 32, 32, 32]
Input Dimension	32
Number of Attention Heads	4
Attention Dropout	0.1
Entropy Regularization Weight	0.003
Aggregate Function	Sum
Message Functions (Complete)	`Complex`, `split`, `Dual`
Layer Normalization	True
Shortcut Connections	True

Table 8: Pre-training hyperparameter configuration for the MERC model.

Checkpoint selection. To select a checkpoint for zero-shot generalization, we used a validation rule based on the three participating pre-training datasets: FB15k-237, WN18RR, and CoDEx-M. Specifically, the final checkpoint was selected according to the highest average MRR over the test splits of these three pre-training datasets. No target zero-shot evaluation graph was used for checkpoint selection.

E Standard Benchmark Datasets

This appendix summarizes the public knowledge graph datasets used for pre-training and zero-shot evaluation. We use 57 publicly available KG datasets spanning transductive, inductive entity-level and inductive entity-and-relation-level settings. Among them, FB15k-237, WN18RR, and CoDEx-M are used exclusively for pre-training, while the remaining 54 graphs are held out for zero-shot evaluation.

Pre-training graphs. The pre-training corpus consists of three standard knowledge graphs: FB15k-237, WN18RR, and CoDEx-M. These graphs provide structurally diverse relational evidence during training but are excluded from the zero-shot test suite. They are used only for learning transferable structural reasoning behavior and for checkpoint selection.

Dataset	Reference	Entities	Relations	Train	Valid	Test	Task
WN18RR	Dettmers et al. (2018)	40943	11	86835	3034	3134	h/t
FB15k-237	Toutanova & Chen (2015)	14541	237	272115	17535	20466	h/t
CoDEx-M	Safavi & Koutra (2020)	17050	51	185584	10310	10311	h/t

Table 9: Three transductive datasets used in the pretraining. Train, valid and test denote the number of triples in the respective set. Task denotes the prediction task at inference: h/t is predicting both heads and tails, t is only predicting tails.

Transductive evaluation graphs. The transductive group contains datasets in which the entity and relation vocabularies are fixed across training and inference within each graph. The full set includes CoDEx-S, CoDEx-L, NELL-995, YAGO310, WD-singer, NELL23K, FB15k-237-10%, FB15k-237-20%, FB15k-237-50%, DB100K, Hetionet, Aristo-V4 and ConceptNet-100K, after excluding the three pre-training graphs from zero-shot evaluation.

Dataset	Reference	Entities	Relations	Train	Valid	Test	Task
CoDEx-S	Safavi & Koutra (2020)	2034	42	32888	1827	1828	h/t
CoDEx-L	Safavi & Koutra (2020)	77951	69	551193	30622	30622	h/t
NELL-995	Xiong et al. (2017)	74536	200	149678	543	2818	h/t
YAGO310	Mahdisoltani et al. (2015)	123182	37	1079040	5000	5000	h/t
WD-singer	Lv et al. (2020)	10282	135	16142	2163	2203	h/t
NELL23K	Lv et al. (2020)	22925	200	25445	4961	4952	h/t
FB15k-237-10%	Lv et al. (2020)	11512	237	27211	15624	18150	t
FB15k-237-20%	Lv et al. (2020)	13166	237	54423	16963	19776	t
FB15k-237-50%	Lv et al. (2020)	14149	237	136057	17449	20324	t
DB100K	Ding et al. (2018)	99604	470	597572	50000	50000	h/t
Hetionet	Himmelstein et al. (2017)	45158	24	2025177	112510	112510	h/t
Aristo-V4	Chen et al. (2021)	44949	1605	242567	20000	20000	h/t
ConceptNet-100K	Malaviya et al. (2020)	78334	34	100000	1200	1200	h/t

Table 10: 13 Transductive datasets used in the evaluation. Train, valid and test denote the number of triples in the respective set. Task denotes the prediction task at inference: h/t is predicting both heads and tails, t is only predicting tails.

Inductive entity-level evaluation graphs. The inductive e group evaluates generalization to unseen entities while keeping the relation vocabulary fixed. This group includes WN18RR v1–v4, FB15k-237 v1–v4, NELL-995 v1–v4, ILPC22-S, ILPC22-L, Hamaguchi-BM 1k, Hamaguchi-BM 3k, Hamaguchi-BM 5k and INDIGO-BM.

Inductive entity-and-relation evaluation graphs. The inductive e, r group evaluates the most challenging zero-shot setting, where both unseen entities and unseen relations appear at inference time. This group includes FB25, FB50, FB75, FB100, WK25, WK50, WK75, WK100, NL0, NL25, NL50, NL75, NL100, WikiTopics-MT1 Tax, WikiTopics-MT1 Health, WikiTopics-MT2 Org, WikiTopics-MT2 Sci, WikiTopics-MT3 Art, WikiTopics-MT3 Infra, WikiTopics-MT4 Sci, WikiTopics-MT4 Health, MetaFam and FBNELL.

Together, these 54 held-out graphs provide a heterogeneous evaluation suite covering encyclopedic, lexical, commonsense, biomedical, taxonomic and synthetic benchmark domains, as well as multiple zero-shot generalization regimes.

F Relational Pattern Mining

This appendix describes the construction of the relation-pattern evaluation subsets used in our structural analysis, following prior knowledge graph pattern analysis practice (Ali et al., 2022). We focus on three relational patterns: *symmetry*, *antisymmetry* and *composition*. These patterns are mined at the relation

Dataset	Reference	Relations	Training Graph		Validation Graph			Test Graph		
			Entities	**Triples**	**Entities**	**Triples**	Q_{valid}	**Entities**	**Triples**	Q_{test}
WN18RR:v1	Teru et al. (2020)	9	2746	5410	2746	5410	630	922	1618	373
WN18RR:v2	Teru et al. (2020)	10	6954	15262	6954	15262	1838	2757	4011	852
WN18RR:v3	Teru et al. (2020)	11	12078	25901	12078	25901	3097	5084	6327	1143
WN18RR:v4	Teru et al. (2020)	9	3861	7940	3861	7940	934	7084	12334	2823
FB15k-237:v1	Teru et al. (2020)	180	1594	4245	1594	4245	489	1093	1993	411
FB15k-237:v2	Teru et al. (2020)	200	2608	9739	2608	9739	1166	1660	4145	947
FB15k-237:v3	Teru et al. (2020)	215	3668	17986	3668	17986	2194	2501	7406	1731
FB15k-237:v4	Teru et al. (2020)	219	4707	27203	4707	27203	3352	3051	11714	2840
NELL-995:v1	Teru et al. (2020)	14	3103	4687	3103	4687	414	225	833	201
NELL-995:v2	Teru et al. (2020)	88	2564	8219	2564	8219	922	2086	4586	935
NELL-995:v3	Teru et al. (2020)	142	4647	16393	4647	16393	1851	3566	8048	1620
NELL-995:v4	Teru et al. (2020)	76	2092	7546	2092	7546	876	2795	7073	1447
ILPC22-S	Galkin et al. (2022)	48	10230	78616	6653	20960	2906	6653	20960	2902
ILPC22-L	Galkin et al. (2022)	65	46626	202446	29246	77044	10179	29246	77044	10184
Hamaguchi-BM:1k	Hamaguchi et al. (2017)	11	36237	93364	36311	93364	1771	9899	18638	476
Hamaguchi-BM:3k	Hamaguchi et al. (2017)	11	32118	71097	32250	71097	1201	19218	38285	1349
Hamaguchi-BM:5k	Hamaguchi et al. (2017)	11	28601	57601	28744	57601	900	23792	48425	2124
INDIGO-BM	Liu et al. (2021)	229	12721	121601	12797	121601	14121	14775	250195	14904

Table 11: 18 Inductive e datasets used in the experiments. Triples denotes the number of edges in the corresponding graph, Q_{valid} denotes the number of queries in the validation set and Q_{test} denotes the number of queries in the test set.

Dataset	Reference	Training Graph			Validation Graph				Test Graph			
		Entities	**Relations**	**Triples**	**Entities**	**Relations**	**Triples**	Q_{valid}	**Entities**	**Relations**	**Triples**	Q_{test}
FB-100	Lee et al. (2023)	4659	134	62809	2624	77	6987	2329	2624	77	6987	2329
FB-50	Lee et al. (2023)	5190	153	85375	4445	205	11636	3879	4445	205	11636	3879
FB-75	Lee et al. (2023)	4659	134	62809	2792	186	9316	3106	2792	186	9316	3106
FB-25	Lee et al. (2023)	5190	163	91571	4097	216	17147	5716	4097	216	17147	5716
WK-100	Lee et al. (2023)	9784	67	49875	12136	37	13487	4496	12136	37	13487	4496
WK-50	Lee et al. (2023)	12022	72	82481	9328	93	9672	3224	9328	93	9672	3225
WK-75	Lee et al. (2023)	6853	52	28741	2722	65	3430	1143	2722	65	3430	1144
WK-25	Lee et al. (2023)	12659	47	41873	3228	74	3391	1130	3228	74	3391	1131
NL-100	Lee et al. (2023)	1258	55	7832	1709	53	2378	793	1709	53	2378	793
NL-75	Lee et al. (2023)	2607	96	11058	1578	116	1818	606	1578	116	1818	607
NL-50	Lee et al. (2023)	4396	106	17578	2335	119	2576	859	2335	119	2576	859
NL-25	Lee et al. (2023)	4396	106	17578	2146	120	2230	744	2146	120	2230	744
NL-0	Lee et al. (2023)	1814	134	7796	2026	112	2287	763	2026	112	2287	763
WikiTopics-MT1:Tax	Zhou et al. (2023)	10000	10	17178	10000	10	17178	1908	10000	9	16526	1834
WikiTopics-MT1:Health	Zhou et al. (2023)	10000	7	14371	10000	7	14371	1596	10000	7	14110	1566
WikiTopics-MT2:Org	Zhou et al. (2023)	10000	10	23233	10000	10	23233	2581	10000	11	21976	2441
WikiTopics-MT2:Sci	Zhou et al. (2023)	10000	16	16471	10000	16	16471	1830	10000	16	14852	1650
WikiTopics-MT3:Art	Zhou et al. (2023)	10000	45	27262	10000	45	27262	3026	10000	45	28023	3113
WikiTopics-MT3:Infra	Zhou et al. (2023)	10000	24	21990	10000	24	21990	2443	10000	27	21646	2405
WikiTopics-MT4:Sci	Zhou et al. (2023)	10000	42	12576	10000	42	12576	1397	10000	42	12516	1388
WikiTopics-MT4:Health	Zhou et al. (2023)	10000	21	15539	10000	21	15539	1725	10000	20	15337	1703
MetaFam	Zhou et al. (2023)	1316	28	13821	1316	28	13821	590	656	28	7257	184
FBNELL	Zhou et al. (2023)	4636	100	10275	4636	100	10275	1055	4752	183	10685	597

Table 12: 23 Inductive e, r datasets used in the experiments. Triples denotes the number of edges in the corresponding graph, Q_{valid} denotes the number of queries in the validation set and Q_{test} denotes the number of queries in the test set.

level from each benchmark graph and are then used to partition the corresponding test triples into pattern-specific subsets.

Symmetry and antisymmetry. For each relation r, we define its observed ordered entity-pair set as

$$P_r = \{(h, t) \mid (h, r, t) \in \mathcal{G}\}.$$

The reverse pair set is

$$P_r^{-1} = \{(t, h) \mid (h, t) \in P_r\}.$$

The symmetry confidence of relation r is computed as

$$\mathrm{conf}_{\mathrm{sym}}(r) = \frac{|P_r \cap P_r^{-1}|}{|P_r|}.$$

A relation is classified as symmetric if

$$\mathrm{conf}_{\mathrm{sym}}(r) \geq \tau.$$

In contrast, the antisymmetry confidence is defined as the complement of the symmetry confidence:

$$\mathrm{conf}_{\mathrm{anti}}(r) = 1 - \mathrm{conf}_{\mathrm{sym}}(r).$$

A relation is classified as antisymmetric if

$$\mathrm{conf}_{\mathrm{anti}}(r) \geq \tau.$$

In our implementation, the confidence threshold is set to $\tau = 0.97$ by default. Therefore, a relation is treated as symmetric only when nearly all observed ordered pairs also have their reverse counterparts, and it is treated as antisymmetric when almost none of its observed pairs are reciprocated.

Composition. Composition patterns are mined using relation triples of the form

$$r_1(x, y) \wedge r_2(y, z) \Rightarrow r_3(x, z).$$

To reduce unnecessary computation, candidate relation pairs (r_1, r_2) are first pre-filtered by checking whether there exists at least one intermediate entity y such that r_1 has an incoming edge into y and r_2 has an outgoing edge from y. For each candidate pair (r_1, r_2), we generate the left-hand-side composed pair set

$$L_{r_1,r_2} = \{(x, z) \mid \exists y,\ (x, r_1, y) \in \mathcal{G} \wedge (y, r_2, z) \in \mathcal{G}\}.$$

For every possible target relation r_3, the composition confidence is computed as

$$\mathrm{conf}_{\mathrm{comp}}(r_1, r_2 \Rightarrow r_3) = \frac{|L_{r_1,r_2} \cap P_{r_3}|}{|L_{r_1,r_2}|}.$$

For each relation r, we retain the maximum composition confidence over all candidate relation pairs:

$$\mathrm{conf}_{\mathrm{comp}}(r) = \max_{r_1,r_2} \mathrm{conf}_{\mathrm{comp}}(r_1, r_2 \Rightarrow r).$$

A relation is classified as compositional if

$$\mathrm{conf}_{\mathrm{comp}}(r) \geq \tau.$$

This means that relation r is selected if it can be well explained as the resulting relation of at least one high-confidence two-hop relational composition.

Pattern-specific test subset construction. After identifying the relation sets for symmetry, antisymmetry and composition, we partition the test triples according to their relation labels. Given a test triple (h, r, t), the triple is written to the symmetry subset if r belongs to the mined symmetric relation set, to the antisymmetry subset if r belongs to the mined antisymmetric relation set, and to the composition subset if r belongs to the mined compositional relation set. These subsets are not mutually exclusive: a relation can be assigned to more than one pattern category if it satisfies multiple confidence criteria. Empty pattern subsets are not materialized as files.

Implementation note. The mining procedure is deterministic and uses only observed triples in each dataset. It identifies relation-level structural regularities and then evaluates model behavior on the corresponding test triples. The confidence threshold $\tau = 0.97$ is used consistently across the three pattern types. No additional triples are introduced during pattern-subset construction.

G Biomedical Protein-Disease Graph Construction

This appendix describes the construction of the biomedical Protein-Disease graphs used for domain-shift evaluation. For each biomedical source in CKG, we first extract source-specific triples and then apply topology-aware reshaping, optional sparsification for oversized graphs, and difficulty-stratified transductive splitting.

Source-specific triple extraction. We construct one graph for each biomedical source by extracting triples from CKG and writing them as tab-separated (h, r, t) records. Each source graph is processed independently, so that the final benchmark contains multiple biomedical graphs with various relation vocabularies, entity sets, and topological properties.

Topology-aware graph reshaping. For each source graph, we first compute a bipartite index

$$B = \frac{|H \cap T|}{|H \cup T|},$$

where H and T denote the sets of head and tail entities. If $B < 0.15$, the graph is treated as strongly bipartite-like. In this case, we inject bidirectional `CO_OCCURS_WITH` edges between source entities that share the same target. For targets with at most 50 source entities, pairwise co-occurrence edges are added up to 30 injected edges per target. For targets with more than 50 sources, 30 sources are sampled and connected as a bidirectional chain. This step reduces extreme bipartite sparsity while preserving all original triples.

Next, we compute an undirected adjacency set and a topology-based centrality score for each node:

$$C(v) = \deg(v) + \frac{1}{|\mathcal{N}(v)|} \sum_{u \in \mathcal{N}(v)} \deg(u).$$

For each edge (h, r, t), we compute its edge clustering coefficient

$$ECC(h, t) = \frac{|\mathcal{N}(h) \cap \mathcal{N}(t)|}{\min\big(|\mathcal{N}(h)| - 1,\ |\mathcal{N}(t)| - 1\big)}.$$

The edge is assigned one topology tag. If $ECC(h, t)$ exceeds a graph-dependent threshold, the tag is `INTRAMODULAR`. The threshold is 0.3 when the average degree is larger than 5, and 0.1 otherwise. If $ECC(h, t) = 0$ and both endpoints have degree larger than 2, the tag is `CROSSTALK`. All remaining edges are tagged as `SYSTEMIC`.

A direction-flow tag is then assigned by comparing endpoint centralities. If $C(h) > 1.2\, C(t)$, the edge is tagged as `DIVERGENT`, if $C(t) > 1.2\, C(h)$, it is tagged as `CONVERGENT`, otherwise it is tagged as `LATERAL`. The final relation label is

$$r' = r_{\texttt{TOPOLOGY_FLOW}}.$$

Duplicate reshaped triples are removed before output.

Pattern Datasets	Symmetry		Antisymmetry		Composition	
	Patterns	**Triple**	**Patterns**	**Triple**	**Patterns**	**Triple**
FB-100	3	0	180	2258	92	557
FB-75	3	43	186	2874	122	1969
FB-50	3	150	193	3150	118	2433
FB-25	4	173	199	4786	131	3693
WK-100	3	0	96	4138	14	443
WK-75	4	121	77	961	21	471
WK-50	3	61	105	2608	20	666
WK-25	2	33	81	947	10	477
NL-100	0	0	91	460	14	197
NL-75	1	10	137	185	24	197
NL-50	0	0	132	586	28	273
NL-25	2	1	130	549	22	162
NL-0	0	0	112	469	34	283
WikiTopics-MT1:Tax	1	19	9	1815	0	0
WikiTopics-MT1:Health	1	206	6	1360	0	0
WikiTopics-MT2:Org	0	0	10	2422	2	32
WikiTopics-MT2:Sci	1	54	15	1596	1	0
WikiTopics-MT3:Art	0	0	52	3113	5	1422
WikiTopics-MT3:Infra	2	759	23	1559	1	15
WikiTopics-MT4:Sci	0	0	42	1383	0	0
WikiTopics-MT4:Health	2	234	18	1196	1	103
MetaFam	0	0	22	184	26	184
FBNELL	0	0	161	267	19	100
WN18RR:v1	0	0	5	55	0	0
WN18RR:v2	0	0	6	207	0	0
WN18RR:v3	0	0	7	792	0	0
WN18RR:v4	1	3	5	750	0	0
FB15k-237:v1	6	0	157	339	29	115
FB15k-237:v2	1	1	176	751	32	260
FB15k-237:v3	0	0	191	1408	40	529
FB15k-237:v4	1	0	194	2294	37	775
NELL-995:v1	0	0	13	151	0	0
NELL-995:v2	0	0	70	486	15	145
NELL-995:v3	1	1	113	556	26	490
NELL-995:v4	0	0	64	610	21	368
ILPC22-S	0	0	44	2765	8	431
ILPC22-L	0	0	60	9824	17	6317
Hamaguchi-BM:1k	0	0	10	476	0	0
Hamaguchi-BM:3k	0	0	10	1349	0	0
Hamaguchi-BM:5k	0	0	10	2123	0	0
INDIGO-BM	3	5	197	13143	116	8914
CoDEx-S	3	295	38	1532	10	360
CoDEx-L	0	0	64	29610	15	9697
NELL-995	0	0	160	2818	30	745
YAGO310	2	19	30	4839	3	323
WD-singer	1	0	121	1914	47	1536
NELL23K	0	0	168	3526	22	801
FB15k-237-10%	3	50	208	17557	99	10962
FB15k-237-30%	3	68	205	19128	112	12365
FB15k-237-50%	3	73	205	19667	132	12662
DB100K	2	10	433	40341	177	38999
Hetionet	0	0	24	112510	0	0
Aristo-V4	0	0	1495	18963	1435	19934
ConceptNet-100K	0	0	32	1199	6	693

Table 13: Pattern statistics across 54 evaluation datasets. Pattern denotes the frequency of the detected pattern types and triples denotes the number of corresponding triples in test subsets

Sparsification of oversized graphs. For oversized source graphs, we apply a topology-aware sparsification step. Given n nodes and m edges, the target number of retained edges is

$$m^\star = \min\left(m,\, \max\left(\frac{15n}{2},\, 0.25m,\, \frac{3n}{2}\right)\right).$$

This rule combines a target average degree of 15, a 25% Pareto-style retention floor, and a minimum connectivity safety bound of $1.5n$ edges.

The sparsifier first constructs a connectivity skeleton by retaining, for each node, the incident edge connected to the highest-degree neighbor. This reduces entity loss during downsampling. The remaining edge budget is allocated proportionally across relation types. Within each relation, candidate edges are ranked by

$$S(h,t) = \log\big(1 + \deg(h)\big)\log\big(1 + \deg(t)\big),$$

and the highest-scoring edges are retained. Graphs that already satisfy the target size are copied without sparsification.

Difficulty-stratified splitting. Each processed source graph is split into train, validation and test sets with ratios 0.8/0.1/0.1 using random seed 2026. The split is transductive: every entity and every relation is first guaranteed to appear in the training set by greedily assigning coverage triples to training.

For the remaining candidate triples, we compute a difficulty score

$$D(h,t) = \left[\log\big(1 + \deg(h) + 1.718\big)\log\big(1 + \deg(t) + 1.718\big)\right]^{-1}.$$

Candidate triples are sorted by decreasing difficulty. The hardest 80% form the evaluation pool, while the easiest 20% are added to training. Validation and test triples are sampled from the evaluation pool using motif-aware coherent sampling: after selecting a seed edge, one adjacent edge sharing either endpoint is preferentially added when available. All unused evaluation pool triples are returned to training.

Summary statistics of the resulting 31 biomedical graphs are provided in Table 14.

H Relation Embedding Visualization

This appendix provides qualitative relation embedding visualizations used as supporting evidence for relation separation. These visualizations are not treated as independent proof of representation disentanglement, but they help illustrate how different models organize relation representations during zero-shot inference.

Visualization protocol. We extract relation embeddings during zero-shot inference on NELL-995 for both Trix and MERC. For each model, we compute pairwise cosine similarities among the selected relations and visualize the resulting similarity matrices as heatmaps. The same relation ordering is used for both models to ensure direct visual comparability.

NELL-995 heatmap comparison. The complete NELL-995 relation-similarity heatmaps are shown in Figures 6 and 5. The comparison is intended to provide qualitative evidence on whether MERC produces more separated relation representations than Trix under the same zero-shot evaluation setting.

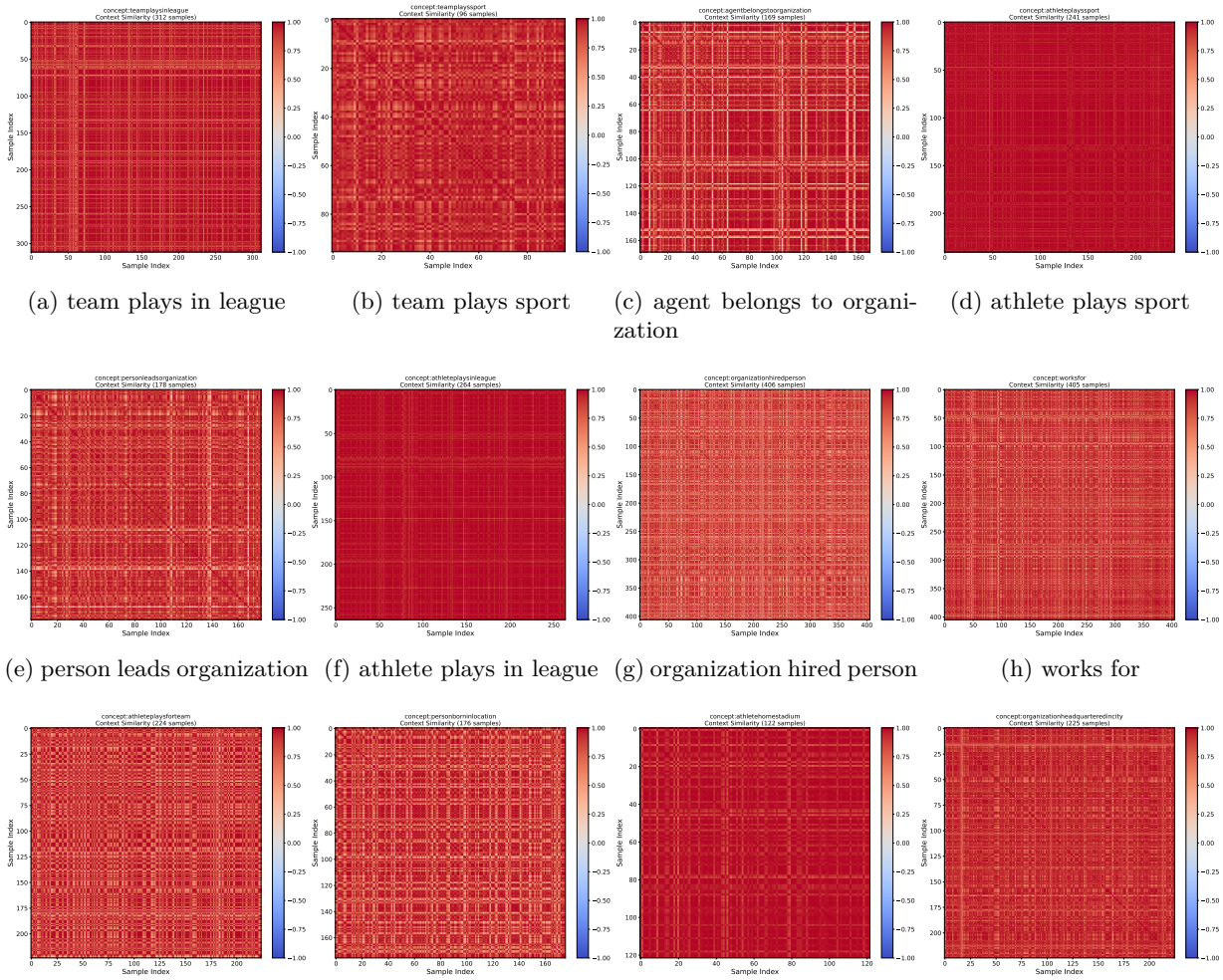
Interpretation. Lower off-diagonal similarity and clearer block separation suggest that relation embeddings are less collapsed and more discriminative.

I Scalability and Profiling Details

This appendix reports the profiling protocol used for scalability analysis, including the special evaluation settings required by memory-intensive baselines.

Dataset	Reference	Entities	Relations	Train	Valid	Test	Task
Cancer Genome Interpreter	Tamborero et al. (2018)	6232	31	9434	1179	1179	h/t
COMPARTMENTS	Binder et al. (2014)	22211	28	744158	10049	10049	h/t
CORUM	Giurgiu et al. (2019)	7069	30	12592	1573	1573	h/t
Comparative Toxicogenomics Database	Davis et al. (2019)	880	31	1706	213	213	h/t
DGIdb	Cotto et al. (2018)	4521	50	26332	3291	3291	h/t
Database of Interacting Proteins	Salwinski et al. (2004)	3498	126	4943	617	617	h/t
DISEASES	Pletscher-Frankild et al. (2015)	27368	36	970809	10077	10076	h/t
DisGeNet	Piñero et al. (2015)	21660	38	203343	25417	25417	h/t
DrugBank	Wishart et al. (2018)	4566	27	285519	35689	35689	h/t
PanelApp	Martin et al. (2019)	5185	30	8052	1006	1006	h/t
GWAS Catalog	Buniello et al. (2019)	15227	51	44932	5616	5616	h/t
HPIDb	Ammari et al. (2016)	210	31	261	32	32	h/t
Human Protein Atlas pathology	Pontén et al. (2008)	105725	11	634346	10070	10071	h/t
Interologous Interaction Database	Brown & Jurisica (2005)	373	61	344	43	43	h/t
InnateDB	Breuer et al. (2013)	148	45	131	16	16	h/t
IntAct	Orchard et al. (2014)	24883	89	149259	18657	18657	h/t
MutationDs	Del-Toro et al. (2019)	4994	38	10930	1366	1366	h/t
MINT	Zanzoni et al. (2002)	8848	112	24362	3045	3045	h/t
ORPHANET	Pavan et al. (2017)	3454	24	5757	719	719	h/t
OncoKB	Chakravarty et al. (2017)	1399	39	2853	356	356	h/t
PFam	El-Gebali et al. (2019)	128803	21	152426	19053	19053	h/t
PSYGENET	Gutiérrez-Sacristán et al. (2015)	95	26	346	43	43	h/t
Reactome	Fabregat et al. (2018)	14201	114	81542	10192	10192	h/t
SIGNOR	Licata et al. (2020)	6233	47	17893	2236	2236	h/t
SMPDB	Jewison et al. (2014)	103242	24	619445	10063	10067	h/t
STITCH	Szklarczyk et al. (2016)	19068	112	114366	14295	14295	h/t
STRING	Szklarczyk et al. (2019)	18958	54	314894	39361	39361	h/t
TISSUES	Palasca et al. (2018)	23864	40	1356295	10176	10179	h/t
UniProt	UniProt Consortium (2018)	1156388	68	3343416	10467	10462	h/t
British Heart Foundation - UCL	Orchard et al. (2014)	1046	37	1356	169	169	h/t
MatrixDB	Chautard et al. (2009)	128	31	129	9	15	h/t

Table 14: Statistics of 31 Biomedical datasets used in the experiments. Train, valid and test denote the number of triples in the respective set. Task denotes the prediction task at inference: h/t is predicting both heads and tails.

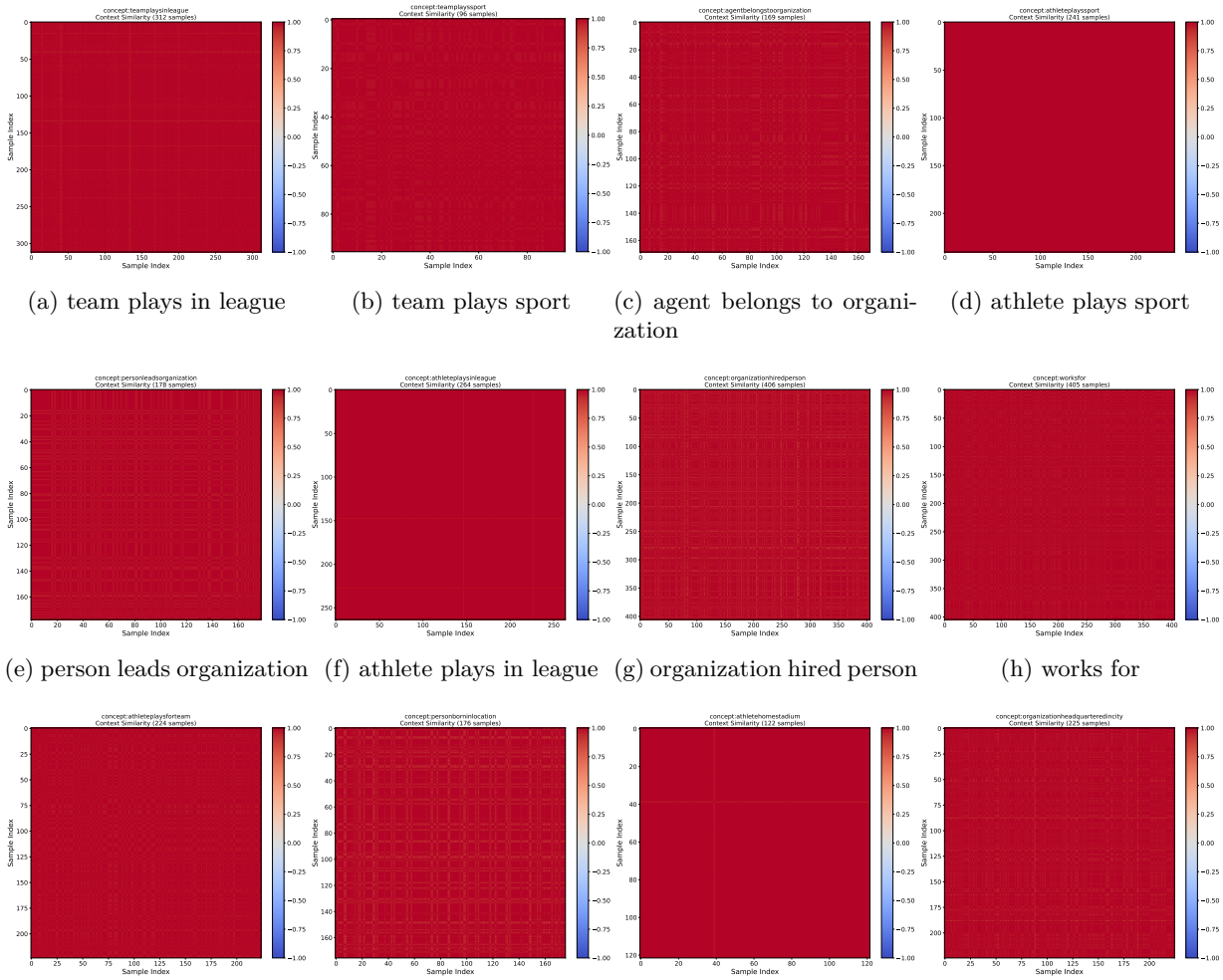


(a) team plays in league (b) team plays sport (c) agent belongs to organization (d) athlete plays sport

(e) person leads organization (f) athlete plays in league (g) organization hired person (h) works for

(i) athlete plays for team (j) person born in location (k) athlete home stadium (l) organization headquartered in city

Figure 5: Heatmaps of cosine similarities of the embeddings of 12 distinct relations during the *zero-shot* inference phase on the NELL-995 dataset for MERC.



(a) team plays in league (b) team plays sport (c) agent belongs to organization (d) athlete plays sport

(e) person leads organization (f) athlete plays in league (g) organization hired person (h) works for

(i) athlete plays for team (j) person born in location (k) athlete home stadium (l) organization headquartered in city

Figure 6: Heatmaps of cosine similarities of the embeddings of 12 distinct relations during the *zero-shot* inference phase on the NELL-995 dataset for Trix.

Flock evaluation settings. For Flock, we use the official evaluation configuration whenever feasible. However, several datasets trigger CPU out-of-memory errors under the default official walk-number settings. In these cases, memory-driven walk-number reductions are applied only when necessary to complete evaluation.

For the relational pattern test sets, the official configuration is used for most benchmarks. DB100K triggers CPU OOM during evaluation, so its `walk_num` is reduced from 512 to 128. CoDEx-L also requires a reduction, with `walk_num` reduced from 512 to 256.

For the custom biomedical graphs, Flock uses the official formulaic heuristic to select `walk_num`. Under this setup, Human Protein Atlas pathology and SMPDB exceed memory capacity, requiring manual walk-number reductions to avoid OOM. The exact `walk_num` and `test_samples` used for each biomedical graph are reported in Table 15.

Dataset	`walk_num`	`test_samples`
MatrixDB	16	16
HPIDb	16	16
British Heart Foundation - UCL	16	16
CORUM	16	16
OncoKB	16	16
ORPHANET	16	16
PanelApp	16	16
Cancer Genome Interpreter	16	16
PSYGENET	16	16
DGIdb	16	16
Interologous Interaction Database	16	16
MutationDs	16	16
Database of Interacting Proteins	16	16
InnateDB	16	16
Comparative Toxicogenomics Database	16	16
SIGNOR	16	16
DrugBank	32	16
MINT	32	16
Human Protein Atlas pathology	64 (512)†	2
Reactome	64	16
GWAS Catalog	64	16
STITCH	64	16
STRING	128	2
IntAct	128	4
DisGeNet	128	4
SMPDB	256 (512)†	2
COMPARTMENTS	256	2
PFam	256	2
TISSUES	256	2
DISEASES	256	2
UniProt	512	2

† Values in parentheses denote the official recommended settings before OOM mitigation.

Table 15: Flock Evaluation Hyperparameters for Custom Topological Biomedical Graphs.

Reporting rule. All scalability results are reported under the actual settings used during evaluation. Memory failures are preserved as part of the scalability comparison, because they indicate practical feasibility limits under the tested hardware environment.

J Dataset-Level Results on Standard Zero-Shot Benchmarks

This appendix reports the complete dataset-level zero-shot link prediction results on the 54 unseen benchmark graphs and training graphs. The results complement the averaged scores in Table 1 and provide the per-dataset evidence behind the aggregate comparison. The individual test results are chronologically listed from 16 to 18.

Dataset	**Ultra**		**Trix**		**MERC Flash**		**MERC**	
	MRR	**Hits@10**	**MRR**	**Hits@10**	**MRR**	**Hits@10**	**MRR**	**Hits@10**
				Inductive e, r				
FB-25	0.388	0.640	0.393	0.650	0.396	0.644	**0.403**	**0.655**
FB-50	0.338	0.543	0.334	0.547	0.339	0.545	**0.345**	**0.554**
FB-75	0.403	0.604	0.401	**0.611**	**0.408**	0.606	**0.408**	**0.611**
FB-100	**0.449**	**0.642**	0.436	0.635	0.441	0.634	0.443	0.641
WK-25	**0.316**	**0.532**	0.305	0.496	0.301	0.499	0.307	0.502
WK-50	0.166	**0.324**	0.166	0.313	0.159	0.297	**0.167**	0.313
WK-75	0.365	0.537	0.368	0.513	0.371	0.510	**0.396**	**0.543**
WK-100	0.164	0.286	**0.188**	0.299	0.176	0.294	0.175	**0.301**
NL-0	0.342	0.523	**0.385**	0.549	0.367	**0.550**	0.357	0.545
NL-25	**0.395**	0.569	0.377	**0.589**	0.365	0.564	0.382	0.534
NL-50	**0.407**	**0.570**	0.404	0.548	0.403	0.565	0.405	0.562
NL-75	**0.368**	**0.547**	0.351	0.525	0.356	0.512	0.346	0.529
NL-100	0.471	0.651	**0.486**	0.676	0.465	0.661	0.451	**0.694**
WikiTopics-MT1:tax	0.224	0.305	0.358	0.452	0.365	0.446	**0.394**	**0.490**
WikiTopics-MT1:health	0.298	0.374	0.376	0.457	0.353	0.454	**0.380**	**0.474**
WikiTopics-MT2:org	0.095	0.159	0.091	0.156	0.094	**0.160**	**0.096**	0.159
WikiTopics-MT2:sci	0.258	0.354	0.323	0.465	0.312	0.453	**0.367**	**0.512**
WikiTopics-MT3:art	0.259	0.402	0.284	0.441	**0.290**	**0.451**	0.281	0.441
WikiTopics-MT3:infra	0.619	0.755	**0.655**	**0.797**	0.651	0.789	0.636	0.780
WikiTopics-MT4:sci	0.274	0.449	0.290	0.460	0.302	**0.478**	**0.303**	0.473
WikiTopics-MT4:health	0.624	0.737	**0.677**	**0.775**	0.616	0.765	0.673	**0.775**
Metafam	0.238	0.644	0.341	0.815	0.694	0.997	**0.757**	**1.000**
FBNELL	0.485	0.652	0.473	0.660	0.497	0.676	**0.511**	**0.689**

Table 16: Zero-shot link prediction performance of Ultra, Trix, MERC Flash and MERC on various inductive e, r datasets.

Dataset	Ultra		Trix		MERC Flash		MERC	
	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
Inductive e								
FB15k-237:v1	0.498	0.656	**0.515**	0.682	0.512	**0.690**	0.507	0.680
FB15k-237:v2	0.512	0.700	0.525	**0.730**	0.515	0.719	**0.530**	0.707
FB15k-237:v3	0.491	0.654	0.501	**0.669**	0.495	0.656	**0.502**	0.661
FB15k-237:v4	0.486	0.677	0.493	**0.687**	0.486	0.680	**0.495**	0.684
WN18RR:v1	0.648	0.768	**0.699**	0.791	0.645	**0.796**	0.669	**0.796**
WN18RR:v2	0.663	0.765	**0.678**	0.781	0.659	0.782	0.667	**0.792**
WN18RR:v3	0.376	0.476	0.418	0.541	**0.451**	0.590	0.450	**0.596**
WN18RR:v4	0.611	0.705	**0.648**	**0.723**	0.592	**0.723**	0.618	**0.723**
NELL-995:v1	0.785	**0.913**	**0.806**	0.898	0.763	0.881	0.672	0.749
NELL-995:v2	0.526	0.707	**0.569**	**0.768**	0.541	0.729	0.561	0.761
NELL-995:v3	0.515	0.702	0.558	0.743	0.570	0.758	**0.582**	**0.780**
NELL-995:v4	0.479	0.712	**0.538**	**0.765**	0.529	0.736	0.536	**0.765**
Hamaguchi-BM:1k	0.059	0.092	0.072	0.128	**0.084**	0.143	0.082	**0.145**
Hamaguchi-BM:3k	0.037	0.077	0.069	0.119	0.068	**0.132**	**0.074**	0.131
Hamaguchi-BM:5k	0.034	0.071	0.062	0.110	0.061	**0.116**	**0.067**	0.113
INDIGO-BM	**0.440**	**0.648**	0.436	0.645	0.416	0.633	0.414	0.632
ILPC2022-S	0.302	0.443	**0.303**	**0.455**	0.302	0.449	0.300	0.451
ILPC2022-L	0.290	0.424	**0.307**	**0.428**	0.297	0.424	0.294	0.420

Table 17: Zero-shot link prediction performance of Ultra, Trix, MERC Flash and MERC on various inductive e datasets.

Dataset	Ultra		Trix		MERC Flash		MERC	
	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
Transductive								
NELL-995	0.406	0.543	0.472	0.629	**0.502**	0.645	0.489	**0.658**
NELL23k	0.239	0.408	**0.290**	**0.497**	0.241	0.400	0.245	0.414
WD-singer	0.382	0.498	**0.511**	**0.609**	0.367	0.497	0.380	0.510
ConceptNet-100k	0.082	0.162	0.193	0.345	**0.291**	**0.458**	0.242	0.436
CoDEx-S	0.472	0.667	0.472	0.670	0.481	0.672	**0.490**	**0.677**
CoDEx-L	0.338	0.469	0.335	0.469	0.340	0.472	**0.343**	**0.479**
YAGO310	0.451	0.615	0.409	0.627	0.444	0.625	**0.510**	**0.684**
Aristo-V4	0.182	0.282	0.181	0.286	**0.258**	0.367	0.252	**0.374**
DB100k	0.398	0.576	0.426	0.603	0.420	0.592	**0.435**	**0.610**
Hetionet	0.257	0.379	**0.279**	**0.420**	0.236	0.373	**0.279**	0.415
FB15k-237-10%	**0.248**	0.398	0.246	0.393	0.244	0.393	0.246	**0.401**
FB15k-237-20%	**0.272**	**0.436**	0.269	0.430	0.265	0.426	0.268	**0.436**
FB15k-237-50%	0.324	**0.526**	0.321	0.521	0.323	0.519	**0.326**	**0.526**
Pretrained								
FB15k-237	0.368	0.564	0.366	0.559	0.371	0.564	**0.375**	**0.569**
WN18RR	0.480	0.614	**0.514**	0.611	0.482	0.625	0.505	**0.632**
CoDEx-M	**0.372**	0.525	0.365	0.521	0.364	0.526	**0.372**	**0.531**

Table 18: Zero-shot link prediction performance of Ultra, Trix, MERC Flash and MERC on various transductive datasets.

K Dataset-Level Results on Relational Pattern and Biomedical Evaluations

This appendix reports the complete dataset-level results for two fine-grained evaluation suites: relational pattern subsets and biomedical Protein-Disease graphs. These results provide the detailed evidence behind Tables 2 and 22.

Dataset	Triple Number	Ultra		Trix		Flock		MERC Flash		MERC	
		MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
FB15k-237:v1	115	0.439	0.570	0.463	0.613	0.445	0.626	0.459	0.596	0.448	0.604
FB15k-237:v2	260	0.439	0.623	0.447	0.677	0.461	0.685	0.452	0.652	0.449	0.629
FB15k-237:v3	529	0.501	0.667	0.524	0.698	0.541	0.709	0.519	0.673	0.515	0.674
FB15k-237:v4	775	0.540	0.683	0.545	0.699	0.553	0.708	0.542	0.692	0.550	0.692
NELL-995:v2	145	0.375	0.555	0.434	0.628	0.451	0.648	0.427	0.624	0.457	0.641
NELL-995:v3	434	0.486	0.636	0.523	0.709	0.596	0.768	0.556	0.747	0.576	0.771
NELL-995:v4	368	0.443	0.592	0.505	0.668	0.493	0.682	0.457	0.582	0.484	0.652
ILPC2022-S	431	0.364	0.517	0.375	0.548	0.385	0.541	0.376	0.544	0.379	0.537
ILPC2022-L	6317	0.326	0.451	0.338	0.457	0.350	0.465	0.326	0.455	0.323	0.450
INDIGO-BM	8914	0.455	0.637	0.459	0.631	0.439	0.628	0.424	0.611	0.418	0.608
FB-25	3693	0.384	0.588	0.393	0.606	0.404	0.621	0.393	0.599	0.400	0.610
FB-50	2433	0.331	0.509	0.342	0.528	0.356	0.547	0.340	0.522	0.348	0.531
FB-75	1969	0.394	0.602	0.407	0.617	0.427	0.634	0.416	0.616	0.413	0.619
FB-100	557	0.374	0.580	0.370	0.590	0.393	0.618	0.383	0.584	0.392	0.596
WK-25	477	0.362	0.523	0.348	0.499	0.354	0.560	0.350	0.509	0.343	0.506
WK-50	666	0.142	0.298	0.157	0.290	0.138	0.280	0.162	0.303	0.168	0.315
WK-75	471	0.380	0.576	0.366	0.545	0.386	0.584	0.365	0.527	0.404	0.599
WK-100	443	0.137	0.297	0.141	0.314	0.181	0.363	0.142	0.312	0.147	0.325
NL-0	283	0.358	0.523	0.443	0.595	0.443	0.633	0.410	0.588	0.395	0.597
NL-25	162	0.381	0.485	0.402	0.617	0.354	0.611	0.424	0.602	0.386	0.491
NL-50	273	0.401	0.533	0.410	0.531	0.382	0.560	0.410	0.548	0.410	0.524
NL-75	197	0.367	0.563	0.352	0.584	0.328	0.579	0.389	0.558	0.391	0.619
NL-100	197	0.427	0.619	0.468	0.685	0.446	0.680	0.438	0.629	0.454	0.695
WikiTopics-MT2:org	32	0.389	0.516	0.427	0.594	0.453	0.625	0.443	0.594	0.423	0.547
WikiTopics-MT3:art	1422	0.312	0.482	0.329	0.489	0.351	0.519	0.339	0.512	0.319	0.484
WikiTopics-MT3:infra	15	0.148	0.333	0.353	0.633	0.365	0.800	0.301	0.600	0.254	0.500
WikiTopics-MT4:health	103	0.197	0.369	0.200	0.369	0.214	0.379	0.186	0.354	0.201	0.379
Metafam	184	0.330	0.821	0.341	0.813	0.484	0.943	0.694	0.997	0.757	1.000
FBNELL	100	0.435	0.645	0.445	0.660	0.469	0.715	0.464	0.680	0.491	0.695
CoDEx-S	360	0.484	0.676	0.428	0.631	0.461	0.649	0.493	0.696	0.503	0.693
CoDEx-L	9697	0.432	0.558	0.408	0.521	0.414	0.512	0.448	0.566	0.451	0.573
NELL-995	745	0.441	0.533	0.456	0.535	0.536	0.620	0.475	0.581	0.478	0.620
YAGO310	323	0.467	0.498	0.467	0.503	0.471	0.505	0.279	0.502	0.297	0.509
ConceptNet-100k	693	0.047	0.085	0.185	0.333	0.249	0.461	0.300	0.468	0.234	0.420
DB100k	38999	0.411	0.575	0.412	0.597	0.430	0.610	0.429	0.599	0.447	0.619
WD-singer	1536	0.334	0.436	0.350	0.455	0.370	0.480	0.331	0.452	0.345	0.461
NELL23k	801	0.278	0.403	0.279	0.398	0.289	0.404	0.285	0.403	0.286	0.414
Aristo-V4	19934	0.182	0.262	0.148	0.258	0.307	0.443	0.258	0.367	0.252	0.374
FB15k-237-10%	10962	0.250	0.387	0.255	0.383	0.259	0.395	0.254	0.387	0.257	0.391
FB15k-237-20%	12365	0.279	0.428	0.276	0.423	0.283	0.440	0.273	0.424	0.276	0.430
FB15k-237-50%	12662	0.344	0.528	0.335	0.521	0.338	0.524	0.343	0.526	0.346	0.533

Table 19: Zero-shot link prediction performance of Ultra, Trix, Flock, MERC Flash and MERC on various composition relational pattern test sets. Triple Number indicates the total number of triples contained in the test set.

Dataset	Triple Number	Ultra		Trix		Flock		MERC Flash		MERC	
		MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
INDIGO-BM	5	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FB-25	173	0.878	0.942	0.992	1.000	1.000	1.000	0.972	0.983	0.976	0.994
FB-50	150	0.944	0.947	0.997	0.997	1.000	1.000	0.953	0.993	0.929	0.947
FB-75	43	0.607	0.733	0.985	1.000	1.000	1.000	0.874	0.953	0.776	0.791
WK-25	33	0.945	1.000	0.958	1.000	0.556	0.864	0.971	0.970	0.928	1.000
WK-50	61	0.844	0.902	0.681	0.885	0.236	0.615	0.842	0.934	0.844	0.926
WK-75	121	0.341	0.455	0.423	0.554	0.397	0.570	0.440	0.537	0.459	0.529
NL-25	1	1.000	1.000	1.000	1.000	0.750	1.000	0.279	0.500	0.500	1.000
NL-75	10	0.850	1.000	0.775	1.000	0.900	1.000	0.875	1.000	0.875	1.000
WikiTopics-MT1:tax	19	0.750	0.868	0.897	0.895	0.897	0.895	0.896	0.895	0.896	0.895
WikiTopics-MT1:health	206	0.658	0.748	0.807	0.818	0.788	0.820	0.661	0.774	0.813	0.828
WikiTopics-MT2:sci	54	0.728	0.815	0.930	0.935	0.931	0.935	0.933	0.935	0.933	0.935
WikiTopics-MT3:infra	759	0.875	0.937	0.914	0.943	0.917	0.953	0.903	0.937	0.905	0.921
WikiTopics-MT4:health	234	0.688	0.865	0.892	0.915	0.850	0.912	0.848	0.919	0.907	0.927
CoDEx-S	295	0.835	0.944	0.558	0.912	0.741	0.924	0.848	0.941	0.859	0.944
YAGO310	19	0.974	1.000	1.000	1.000	0.987	1.000	0.794	1.000	0.952	1.000
DB100k	10	1.000	1.000	0.800	1.000	1.000	1.000	1.000	1.000	1.000	1.000
FB15k-237-10%	50	0.742	0.740	0.977	1.000	1.000	1.000	0.573	0.800	0.582	0.720
FB15k-237-20%	68	0.898	0.897	0.978	1.000	1.000	1.000	0.759	0.868	0.786	0.824
FB15k-237-50%	73	0.993	1.000	1.000	1.000	1.000	1.000	0.957	1.000	0.979	1.000

Table 20: Zero-shot link prediction performance of Ultra, Trix, Flock, MERC Flash and MERC on various symmetry relational pattern test sets. Triple Number indicates the total number of triples contained in the test set.

Dataset	Triple Number	Ultra		Trix		Flock		MERC Flash		MERC	
		MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
FB15k-237:v1	339	0.414	0.591	0.440	0.619	0.424	0.639	0.449	0.631	0.434	0.615
FB15k-237:v2	751	0.436	0.638	0.460	0.680	0.461	0.685	0.458	0.668	0.464	0.654
FB15k-237:v3	2294	0.427	0.589	0.451	0.617	0.460	0.637	0.442	0.604	0.449	0.608
FB15k-237:v4	1408	0.417	0.616	0.434	0.635	0.439	0.649	0.430	0.627	0.436	0.632
WN18RR:v1	55	0.078	0.273	0.111	0.273	0.099	0.273	0.105	0.318	0.108	0.318
WN18RR:v2	207	0.161	0.319	0.161	0.329	0.211	0.372	0.190	0.350	0.184	0.370
WN18RR:v3	750	0.177	0.316	0.230	0.390	0.297	0.472	0.276	0.449	0.275	0.459
WN18RR:v4	792	0.116	0.241	0.122	0.243	0.130	0.263	0.113	0.237	0.123	0.251
NELL-995:v1	171	0.746	0.901	0.817	0.901	0.670	0.865	0.790	0.898	0.666	0.725
NELL-995:v2	486	0.461	0.639	0.504	0.701	0.530	0.726	0.514	0.698	0.511	0.702
NELL-995:v3	610	0.472	0.647	0.531	0.713	0.557	0.739	0.542	0.710	0.541	0.737
NELL-995:v4	556	0.407	0.561	0.487	0.649	0.518	0.680	0.467	0.617	0.490	0.646
ILPC2022-S	9824	0.284	0.423	0.290	0.437	0.297	0.440	0.289	0.431	0.286	0.434
ILPC2022-L	2765	0.284	0.407	0.295	0.413	0.307	0.424	0.285	0.410	0.282	0.405
Hamaguchi-BM:1k	476	0.079	0.150	0.072	0.128	0.068	0.126	0.084	0.143	0.082	0.145
Hamaguchi-BM:3k	1349	0.063	0.120	0.069	0.120	0.067	0.118	0.068	0.132	0.074	0.132
Hamaguchi-BM:5k	2123	0.055	0.102	0.062	0.110	0.064	0.118	0.061	0.116	0.067	0.114
INDIGO-BM	13143	0.427	0.618	0.427	0.614	0.411	0.605	0.406	0.602	0.402	0.600
FB-25	4786	0.355	0.596	0.357	0.607	0.366	0.621	0.361	0.603	0.367	0.615
FB-50	3150	0.320	0.525	0.318	0.532	0.336	0.548	0.326	0.530	0.331	0.538
FB-75	2874	0.391	0.594	0.393	0.604	0.409	0.616	0.403	0.601	0.404	0.610
FB-100	2258	0.434	0.627	0.433	0.632	0.447	0.654	0.434	0.628	0.438	0.637
WK-25	947	0.279	0.473	0.273	0.459	0.261	0.459	0.272	0.466	0.278	0.472
WK-50	2608	0.134	0.268	0.129	0.276	0.111	0.238	0.129	0.262	0.135	0.278
WK-75	961	0.384	0.523	0.367	0.503	0.383	0.528	0.368	0.503	0.393	0.536
WK-100	4138	0.183	0.293	0.193	0.301	0.187	0.298	0.180	0.297	0.178	0.303
NL-0	469	0.358	0.532	0.423	0.590	0.414	0.623	0.398	0.570	0.385	0.568
NL-25	549	0.393	0.559	0.363	0.566	0.364	0.577	0.354	0.536	0.388	0.534
NL-50	586	0.385	0.530	0.402	0.522	0.376	0.540	0.398	0.537	0.405	0.543
NL-75	185	0.427	0.600	0.442	0.611	0.412	0.576	0.450	0.586	0.437	0.600
NL-100	460	0.476	0.670	0.496	0.705	0.469	0.692	0.480	0.671	0.480	0.718
WikiTopics-MT1:tax	1815	0.237	0.299	0.353	0.449	0.272	0.377	0.359	0.442	0.389	0.486
WikiTopics-MT1:health	1360	0.222	0.269	0.312	0.403	0.321	0.430	0.307	0.405	0.314	0.421
WikiTopics-MT2:org	2422	0.081	0.143	0.089	0.152	0.099	0.162	0.091	0.157	0.094	0.156
WikiTopics-MT2:sci	1596	0.242	0.332	0.303	0.449	0.298	0.436	0.291	0.437	0.348	0.497
WikiTopics-MT3:art	3113	0.251	0.414	0.284	0.441	0.300	0.465	0.290	0.451	0.281	0.441
WikiTopics-MT3:infra	1559	0.519	0.720	0.550	0.744	0.592	0.773	0.552	0.734	0.526	0.729
WikiTopics-MT4:sci	1383	0.293	0.455	0.289	0.458	0.302	0.467	0.300	0.476	0.301	0.472
WikiTopics-MT4:health	1196	0.449	0.616	0.575	0.701	0.591	0.712	0.493	0.685	0.562	0.698
Metafam	184	0.330	0.821	0.341	0.813	0.484	0.943	0.694	0.997	0.757	1.000
FBNELL	267	0.438	0.627	0.457	0.622	0.450	0.646	0.447	0.637	0.462	0.637
CoDEx-S	1532	0.402	0.615	0.360	0.581	0.383	0.588	0.410	0.620	0.419	0.626
CoDEx-L	29610	0.316	0.445	0.305	0.434	0.315	0.434	0.323	0.457	0.326	0.463
NELL-995	2818	0.437	0.575	0.463	0.606	0.495	0.656	0.502	0.645	0.489	0.658
YAGO310	4389	0.499	0.681	0.409	0.636	0.413	0.682	0.451	0.630	0.520	0.692
ConceptNet-100k	1199	0.061	0.117	0.190	0.346	0.250	0.449	0.291	0.458	0.242	0.436
DB100k	40341	0.330	0.497	0.340	0.515	0.364	0.523	0.360	0.524	0.371	0.540
WD-singer	1914	0.334	0.437	0.343	0.457	0.360	0.479	0.326	0.446	0.326	0.460
NELL23k	3526	0.251	0.410	0.248	0.398	0.241	0.399	0.252	0.406	0.254	0.408
Aristo-V4	18963	0.187	0.266	0.150	0.261	0.312	0.448	0.262	0.370	0.257	0.376
Hetionet	112510	0.261	0.382	0.196	0.369	0.246	0.371	0.236	0.373	0.279	0.415
FB15k-237-10%	17557	0.244	0.390	0.246	0.394	0.249	0.408	0.248	0.397	0.250	0.406
FB15k-237-20%	19128	0.271	0.431	0.268	0.427	0.276	0.448	0.268	0.429	0.271	0.440
FB15k-237-50%	19667	0.324	0.520	0.316	0.511	0.321	0.519	0.325	0.519	0.329	0.526

Table 21: Zero-shot link prediction performance of Ultra, Trix, Flock, MERC Flash and MERC on various anti-symmetry relational pattern test sets. Triple Number indicates the total number of triples contained in the test set.

Dataset	Triple Number	Ultra		Trix		Flock		MERC Flash		MERC	
		MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
British Heart Foundation - UCL	169	0.389	0.621	0.238	0.536	0.336	0.633	0.388	0.654	0.391	0.654
Cancer Genome Interpreter	1179	0.384	0.575	0.487	0.606	0.697	0.808	0.653	0.749	0.635	0.747
CORUM	1573	0.337	0.572	0.329	0.583	0.412	0.647	0.406	0.623	0.391	0.614
Comparative Toxicogenomics Database	213	0.633	0.866	0.675	0.892	0.760	0.951	0.752	0.887	0.766	0.906
DGIdb	3291	0.343	0.598	0.472	0.729	0.533	0.762	0.458	0.700	0.509	0.754
Database of Interacting Proteins	617	0.228	0.438	0.266	0.475	0.306	0.532	0.254	0.467	0.262	0.438
DisGeNet	25417	0.107	0.164	0.152	0.279	0.256	0.385	0.219	0.358	0.242	0.382
DrugBank	35689	0.089	0.159	0.049	0.104	0.056	0.104	0.078	0.155	0.079	0.153
PanelApp	1006	0.703	0.917	0.695	0.922	0.794	0.955	0.781	0.942	0.788	0.939
GWAS Catalog	5616	0.409	0.622	0.656	0.850	0.741	0.893	0.644	0.829	0.676	0.846
HPIDb	32	0.276	0.625	0.271	0.641	0.340	0.703	0.323	0.719	0.296	0.609
Interologous Interaction Database	43	0.208	0.384	0.272	0.488	0.348	0.709	0.283	0.535	0.197	0.337
InnateDB	16	0.284	0.500	0.232	0.406	0.275	0.594	0.351	0.594	0.293	0.531
IntAct	18657	0.014	0.027	0.031	0.066	0.052	0.117	0.082	0.170	0.078	0.157
MutationDs	1366	0.555	0.842	0.656	0.906	0.692	0.930	0.686	0.896	0.703	0.902
MatrixDB	15	0.174	0.367	0.208	0.467	0.264	0.567	0.145	0.233	0.210	0.400
MINT	3045	0.069	0.144	0.114	0.232	0.133	0.272	0.131	0.251	0.149	0.276
OncoKB	356	0.540	0.742	0.561	0.743	0.885	0.980	0.834	0.909	0.776	0.897
ORPHANET	719	0.756	0.925	0.785	0.961	0.865	0.981	0.851	0.975	0.853	0.972
PFam	19053	0.259	0.450	0.402	0.570	0.488	0.689	0.472	0.698	0.499	0.656
PSYGENET	43	0.690	0.884	0.636	0.930	0.744	0.977	0.699	0.872	0.740	0.919
Reactome	10192	0.277	0.508	0.448	0.709	0.514	0.782	0.459	0.702	0.483	0.743
SIGNOR	2236	0.508	0.724	0.571	0.742	0.654	0.801	0.590	0.745	0.621	0.787
STITCH	14295	0.257	0.422	0.280	0.500	0.450	0.647	0.424	0.637	0.426	0.639
STRING	39361	0.089	0.160	0.076	0.152	0.157	0.286	0.181	0.317	0.194	0.327
COMPARTMENTS	10049	0.045	0.075	0.063	0.119	0.078	0.150	0.069	0.140	0.078	0.157
DISEASES	10076	0.043	0.084	0.062	0.122	0.069	0.122	0.051	0.099	0.060	0.120
Human Protein Atlas pathology	10071	0.137	0.492	0.170	0.499	0.224	0.498	0.236	0.500	0.190	0.498
SMPDB	10067	0.297	0.418	0.332	0.471	0.382	0.486	0.331	0.453	0.345	0.464
TISSUES	10179	0.026	0.045	0.046	0.080	0.047	0.078	0.046	0.092	0.042	0.083
UniProt	10462	0.192	0.283	0.477	0.688	0.536	0.777	0.487	0.682	0.481	0.674

Table 22: Zero-shot link prediction performance of Ultra, Trix, Flock, MERC Flash and MERC on various biomedical datasets. Triple Number indicates the total number of triples contained in the test set.