HDiff: Confidence-Guided Denoising Diffusion for Robust Hyper-relational Link Prediction

Xiangfeng Luo and Ruoxin Zheng and Jianqiang Huang and Hang Yu*
School of Computer Engineering and Science, Shanghai University
Shanghai 200444, China
{luoxf, zhengruoxin, huangjq, yuhang}@shu.edu.cn

Abstract

Although Hyper-relational Knowledge Graphs (HKGs) can model complex facts better than traditional KGs, the Hyper-relational Knowledge Graph Completion (HKGC) is more sensitive to inherent noise, particularly struggling with two prevalent HKG-specific noise types: *Intra-fact Inconsistency* and *Cross-fact Association Noise*. To address these challenges, we propose **HDiff**, a novel conditional denoising diffusion framework for robust HKGC that learns to reverse structured noise corruption. HDiff integrates a **Consistency-Enhanced Global Encoder (CGE)** using contrastive learning to enforce intra-fact consistency and a **Context-Guided Denoiser (CGD)** performing iterative refinement. The CGD features dual conditioning leveraging CGE's global context and local confidence estimates, effectively combatting both noise types. Extensive experiments demonstrate that HDiff substantially outperforms state-of-the-art HKGC methods, highlighting its effectiveness and significant robustness, particularly under noisy conditions.

1 Introduction

Hyper-relational Knowledge Graphs (HKGs) (Hogan et al., 2021; Xiong et al., 2023) extend traditional KGs (Bordes et al., 2013) by augmenting core facts (s, r, o) with qualifier sets $Q = \{(q_i, v_i)\}$ (Vrandečić and Krötzsch, 2014). This richer structure captures complex factual statements with crucial context, vital for applications like detailed question answering and biomedical discovery (Yan et al., 2022). Hyper-relational Knowledge Graph Completion (HKGC)—predicting missing elements in these hyper-relational facts (H-Facts)—is thus essential for enriching these knowledge bases.

Early HKGC approaches often adapted traditional KG embedding techniques, utilizing compositional scoring functions to integrate qualifier information (Guan et al., 2019; Rosso et al., 2020). More recently, methods employing Graph Neural Networks (GNNs) have gained prominence, leveraging the inherent graph or hypergraph structure of HKGs (Fatemi et al., 2022; Xu et al., 2021). These GNN-based models typically feature specialized message-passing schemes or attention mechanisms designed to effectively aggregate information from both core triple components and associated qualifiers (Luo et al., 2023; Xu et al., 2021). By capturing intricate structural dependencies and learning expressive representations, these methods have significantly advanced the state of the art in HKGC. Nevertheless, their success largely hinges on the assumption of relatively clean data, often neglecting the pervasive issue of noise commonly found in real-world knowledge graphs.

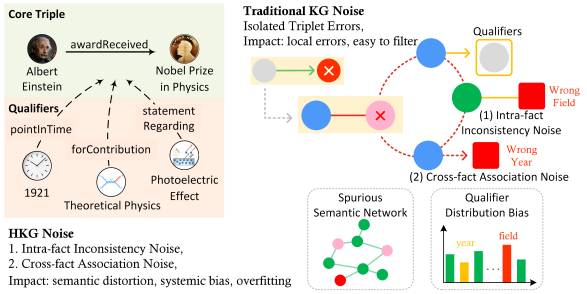

Figure 1: Conceptual comparison of noise types. Left: Structure of a hyper-relational fact (H-Fact). Right: Traditional KG noise contrasted with HKG-specific noise stemming from qualifiers.

Crucially, the complex structure of H-Facts, particularly the interplay involving qualifiers, introduces distinct noise patterns that pose unique challenges beyond those encountered in simpler triple-based KGs (see Figure 1). We identify two prevalent and particularly detrimental HKG-specific noise types arising from these qualifier interactions: (1) **Intra-fact Inconsistency Noise**: This refers to contradictions or inconsistencies arising *within* a single H-Fact, often involving conflicting informa-

*Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 7417–7434
November 4-9, 2025

tion between the core triple and its qualifiers, or among the qualifiers themselves. For example, an H-Fact might state someone received an award in a specific year (qualifier), while the core triple implies a context inconsistent with that timeframe. This internal incoherence, driven by semantically conflicting components within the fact, compromises the integrity of individual factual units. (2) **Cross-fact Association Noise**: This noise manifests as spurious semantic links or systemic contradictions established *between* distinct H-Facts, often propagated via misleading or inaccurate qualifiers. For instance, inconsistent qualifiers across multiple H-Facts describing related events could create misleading relational patterns or erroneous associations between entities, distorting the graph's broader structural representation. This type of noise leverages qualifier semantics to propagate errors in ways distinct from simple incorrect links in traditional KGs.

These specific noise patterns severely undermine the effectiveness of existing HKGC methods (Fatemi et al., 2022; Luo et al., 2023). Methods relying on aggregation (like GNNs) may propagate inconsistencies, while those using scoring functions might assign high confidence to internally contradictory facts. Standard noise mitigation techniques developed for traditional KGs (Hong et al., 2020; Tenorio et al., 2023) often prove insufficient as they are typically not designed to handle inconsistencies stemming from the complex interplay of qualifiers within and across H-Facts.

To address these challenges, we propose **HDiff**, a novel conditional denoising diffusion framework (Ho et al., 2020) for robust HKGC. HDiff adopts a generative approach, learning to reverse a structured noise corruption process. It integrates two synergistic components: (1) A **Consistency-Enhanced Global Encoder (CGE)** employs contrastive learning to enforce intra-fact semantic coherence, directly counteracting *Intra-fact Inconsistency Noise*. (2) A **Context-Guided Denoiser (CGD)** performs iterative refinement conditioned on both global context (from CGE) and local component confidence estimates, robustly mitigating *both* noise types by balancing global structure and local uncertainty. Therefore, our main contributions are threefold:

- We identify and characterize two critical noise patterns prevalent in HKGs: *Intra-fact Inconsistency Noise* and *Cross-fact Association Noise*, highlighting the unique challenges posed by qualifier interactions in their manifestation and impact.
- We propose HDiff, a novel conditional diffusion framework for robust HKGC, featuring:
  - A **Consistency-Enhanced Global Encoder (CGE)** employing targeted contrastive learning (comparing the full H-Fact representation against its core triple representation) to explicitly enforce intra-fact semantic consistency, thereby counteracting *Intra-fact Inconsistency Noise*.
  - A **Context-Guided Denoiser (CGD)** featuring a dual conditioning mechanism (global context + local confidence estimates) to robustly reconstruct H-Facts, effectively addressing both *Intra-fact Inconsistency Noise* and *Cross-fact Association Noise*.
- We conduct extensive experiments on benchmark HKGC datasets under various synthetic and realistic noise settings. Our results demonstrate that HDiff achieves state-of-the-art performance and demonstrates significantly enhanced robustness compared to existing methods, particularly in high-noise scenarios.

2 Related Work

2.1 Hyper-relational Knowledge Graph Completion

Hyper-relational Knowledge Graph Completion (HKGC) focuses on predicting missing elements within hyper-relational facts (H-Facts) (s, r, o, Q), where qualifiers $Q = \{(q_i, v_i)\}$ provide crucial context to the core triple (s, r, o) (Vrandečić and Krötzsch, 2014; Hogan et al., 2021). Early works often decomposed H-Facts or used compositional functions to integrate qualifier information (Wen et al., 2016a; Guan et al., 2019; Liu et al., 2020; Rosso et al., 2020). While these methods successfully integrated qualifiers, they typically lack explicit mechanisms to ensure internal consistency, making them susceptible to **Intra-fact Inconsistency Noise**, where contradictory information within a single H-Fact can significantly degrade performance.

More recent approaches leverage Graph Neural Networks (GNNs) or attention mechanisms to model the complex structure of HKGs (Fatemi

et al., 2022; Xu et al., 2021; Luo et al., 2023). For instance, GRAN (Xu et al., 2021) uses graph attention for information aggregation, while HAHE (Luo et al., 2023) employs hierarchical attention to weigh H-Fact components. Although powerful on clean data, these models often implicitly assume data integrity. Standard GNN aggregation can inadvertently propagate errors stemming from misleading qualifiers (**Cross-fact Association Noise**), and attention mechanisms may struggle to correctly down-weight inconsistent components within an H-Fact (**Intra-fact Inconsistency Noise**), especially when learned representations are distorted by noise. A common limitation across most existing HKGC methods is the lack of explicit, intrinsic robustness mechanisms specifically designed to counteract these prevalent, qualifier-driven noise patterns.

2.2 Noise Robustness in Knowledge Graph Completion

Addressing noise is critical for reliable Knowledge Graph Completion (KGC) (Dong et al., 2025). In traditional triple-based KGs, robustness techniques include noise filtering (Borrego et al., 2019), robust training objectives or sampling strategies (Sun et al., 2019), rule-based consistency enforcement (Tang et al., 2024), and robust GNN aggregators (Tenorio et al., 2023). These methods, however, primarily address triple-level noise (e.g., incorrect links) and are insufficient for HKG noise rooted in the complex semantic interplay of qualifiers within and across H-Facts.

Research on robust HKGC is notably less developed. Adapting methods for noisy traditional KGs like IDKG (Hong and Ma, 2023) is challenging due to the fundamentally different nature of hyper-relational noise. While methods like NYLON (Yu et al., 2024) explore handling noisy H-Facts using confidence, they rely on active learning or external annotations for noise identification, requiring costly manual effort or supervision. This highlights a significant gap: the need for HKGC methods with intrinsic robustness mechanisms operating without external supervision or noise labeling.

Recently, diffusion models have emerged in the KG domain. For instance, DiffKG (Jiang et al., 2024) refines preference scores for recommendation, while KGDM (Long et al., 2024) uses unconditional diffusion for triple completion (see Table 1). However, these models operate on standard triples or user-item matrices. In contrast, HDiff is the first to employ a dual-conditional diffusion process directly on hyper-relational entity and relation representations. This allows it to address the unique structural complexities and noise patterns inherent to HKGs, representing a distinct advancement for robust link prediction in this setting.

Work	Task	Conditional	Handles HKGs/Noise?	Key Difference from Our Work
DiffKG(Jiang et al., 2024)	Recommendation	✗	✗	Re-ranks nodes; Diffusion on 2D interaction matrix to refine preference scores.
KGDM(Long et al., 2024)	Triple KGE	✗	✗	Unconditional diffusion to refine score distribution for triples.
HDiff (Ours)	HKG Completion	✓ local & global	✓ two types	Dual-conditional diffusion on entity/relation reps. for robust HKG completion.

Table 1: Comparison with recent diffusion-based KG models. HDiff is uniquely designed for the complexities of robust HKG completion with a novel dual-conditional mechanism.

Motivated by the distinct challenges posed by HKG-specific noise patterns and the limitations of existing methods regarding intrinsic robustness, we propose HDiff, a novel conditional denoising diffusion framework designed for robust HKGC. In contrast to prior art which often assumes clean data, lacks explicit noise handling mechanisms, or requires external noise labels, HDiff incorporates explicit, learnable components to address noise.

3 Methodology

We propose **HDiff**, a novel conditional denoising diffusion framework specifically designed for robust Hyper-relational Knowledge Graph Completion (HKGC). HDiff addresses HKG noise via two core components: a **Consistency-Enhanced Global Encoder (CGE)** generating noise-resilient, semantically consistent initial embeddings, and a **Context-Guided Denoiser (CGD)** for iterative refinement via conditional diffusion, guided by local and global context for accurate, robust completion. Figure 2 provides a schematic overview of the proposed HDiff architecture.

3.1 Consistency-Enhanced Global Encoder (CGE)

The CGE module (depicted in the upper panel of Figure 2) aims to produce initial embeddings for HKG components (entities, relations, and H-Facts) that are robust to noise, particularly counteracting Intra-fact Inconsistency.

First, we utilize a Graph Attention Network (GAT) (Veličković et al., 2017) applied over the input HKG structure to generate intermediate context-aware embeddings ($\mathbf{z}_k$) for entities, relations, and

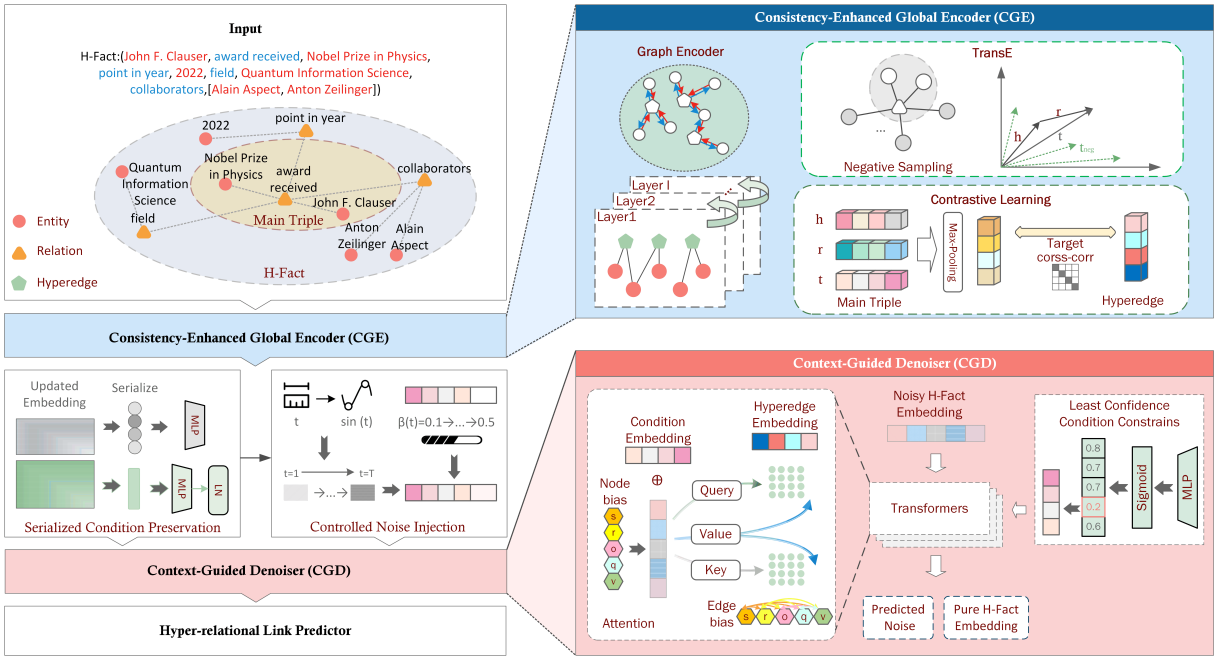

Figure 2: Overview of HDiff. CGE's robust embeddings are prepared for CGD: H-Fact sequences undergo Controlled Noise Injection (to $\mathbf{x}_t$), while Serialized Condition Preservation derives conditioning signals ($\mathbf{c}_{lc}, \mathbf{c}_{he}$).

H-Facts. We chose GAT for its ability to dynamically assign importance weights to different components, including qualifiers of varying significance. This flexibility proved crucial for robustness, as GAT empirically outperformed other encoders like GCN in noisy settings (see Appendix D.1). While GATs are effective at capturing graph structure through message passing, standard aggregation mechanisms are prone to propagating noise present in the graph.

To explicitly combat Intra-fact Inconsistency, we integrate a contrastive learning objective based on Barlow Twins (BT) (Zbontar et al., 2021). This objective enforces semantic coherence by encouraging alignment between two key representations derived from the GAT outputs: the aggregate representation of the full H-Fact ($\mathbf{z}_f$), and a representation focused solely on its core triple ($\mathbf{z}_{sro}$). The GAT yields $\mathbf{z}_f$ as the aggregated representation for the entire H-Fact and component embeddings $\mathbf{z}_k$ for all components (entities, relations, qualifiers). We derive the core triple representation $\mathbf{z}_{sro}$ by applying max pooling over the subject, relation, and object embeddings: $\mathbf{z}_{sro} = \text{MaxPooling}(\mathbf{z}_s, \mathbf{z}_r, \mathbf{z}_o)$. Both $\mathbf{z}_f$ and $\mathbf{z}_{sro}$ are then passed through a shared projector $\mathcal{P}_{\text{BT}}$ to obtain projected representations $\mathbf{p}_f$ and $\mathbf{p}_{sro}$. The Barlow Twins loss $\mathcal{L}_{BT}$ minimizes the redundancy between the dimensions of these projected representations by forcing their empirical cross-correlation matrix $\mathcal{C}$ towards the identity matrix $\mathbf{I}$:

$$\mathcal{L}_{BT} = \sum_i (1 - \mathcal{C}_{ii})^2 + \beta_{BT} \sum_i \sum_{j \neq i} \mathcal{C}_{ij}^2 \quad (1)$$

where β_{BT} is a balancing hyperparameter. By encouraging the full H-Fact representation to be predictable from its core triple representation (and vice versa), $\mathcal{L}_{BT}$ effectively regularizes the embeddings, making them less susceptible to internal contradictions caused by inconsistent or misleading qualifiers within a single H-Fact.

Additionally, to anchor the learned embeddings in fundamental relational semantics and provide a basic structural signal, we apply the TransE (Bordes et al., 2013) margin-based ranking loss ($\mathcal{L}_{TransE}$) to core triples (s, r, o). We use standard negative sampling by corrupting the tail entity to produce negative triples (s, r, o'):

$$\mathcal{L}_{TransE} = \sum_{(s,r,o) \in \mathcal{F}_{\text{core}}} \sum_{o' \in \mathcal{N}'_o(s,r)} \Big[\gamma + \|\mathbf{z}_s + \mathbf{z}_r - \mathbf{z}_o\|_2 - \|\mathbf{z}_s + \mathbf{z}_r - \mathbf{z}_{o'}\|_2\Big]_+^{(2)}$$

Here, $\mathbf{z}_k$ denotes the intermediate embedding from the GAT for component $k \in \{s, r, o, o'\}$, $\|\cdot\|_2$ is the L2 norm, γ is the margin, $\mathcal{F}_{\text{core}}$ is the set of core triples, and $\mathcal{N}'_o(s, r)$ is the set of corrupted tails for (s, r, o).

Finally, an MLP adaptation layer g_ψ transforms the intermediate embeddings $\mathbf{z}_k$ into final adapted embeddings $\mathbf{e}_k$ of dimension d: $\mathbf{e}_k = g_\psi(\mathbf{z}_k)$. The total training loss for the CGE module is a weighted sum of the two objectives:

$$\mathcal{L}_{CGE} = \lambda_{TransE}\mathcal{L}_{TransE} + \lambda_{BT}\mathcal{L}_{BT} \quad (3)$$

where λ_{TransE} and λ_{BT} are hyperparameters balancing the contribution of each loss. This comprehensive process yields robust and semantically consistent initial embeddings for entities ($\mathbf{E}_E$), relations ($\mathbf{E}_R$), and H-Facts ($\mathbf{E}_H$), providing the crucial input for the subsequent denoising stage.

3.2 Context-Guided Denoiser (CGD)

The CGD employs a conditional Denoising Diffusion Probabilistic Model (DDPM) framework (Ho et al., 2020) operating on robust CGE embeddings to iteratively reconstruct clean H-Fact representations, thereby mitigating both *Intra-fact Inconsistency* and *Cross-fact Association Noise*. As shown in the lower panel of Figure 2, for this diffusion process, each H-Fact $f = (s, r, o, \{(k_i, v_i)\}_{i=1}^n)$ is serialized into an initial sequence of component embeddings $\mathbf{x}_0 \in \mathbf{R}^{L\times d}$ by concatenating their corresponding final CGE embeddings:

$$\mathbf{x}_0 = [\mathbf{e}_s, \mathbf{e}_r, \mathbf{e}_o, \mathbf{e}_{k_1}, \mathbf{e}_{v_1}, \ldots, \mathbf{e}_{k_n}, \mathbf{e}_{v_n}] \quad (4)$$

where $L = 3 + 2n$ is the length of the sequence. The indices i and j used hereafter refer to positional indices within this sequence; for instance, $i = 1$ corresponds to the subject embedding $\mathbf{e}_r$.

We apply the standard DDPM forward noising process, which gradually adds Gaussian noise over T timesteps to $\mathbf{x}_0$, producing noisy states $\mathbf{x}_t$. Training involves optimizing a Transformer-based denoiser $\boldsymbol{\epsilon}_\theta$ to predict the added noise $\boldsymbol{\epsilon}$ from the noisy state $\mathbf{x}_t$ and the current timestep t.

To enhance robustness against specific HKG noise patterns, the denoiser $\boldsymbol{\epsilon}_\theta$ is guided by HDiff's dual conditioning mechanism using CGE outputs:

(1) Local Confidence Condition ($\mathbf{c}_{lc}$): To specifically combat *Intra-fact Inconsistency* at a fine-grained level, we identify the component embedding $\mathbf{e}_j$ within the initial $\mathbf{x}_0$ that is predicted to have the lowest confidence score by a trainable MLP $\mathcal{P}_{\text{conf}}$:

$$j = \text{argmin}_{i\in\{1,\ldots,L\}}\{\text{sigmoid}(\mathcal{P}_{\text{conf}}(\mathbf{e}_i))\} \quad (5)$$

This component $\mathbf{e}_j$ corresponds to the element within the H-Fact most likely to be erroneous or inconsistent based on the learned confidence scores. The embedding $\mathbf{e}_j$ is then transformed into the local confidence condition vector $\mathbf{c}_{lc}$ (e.g., via projection). This $\mathbf{c}_{lc}$ is subsequently concatenated as an extra token to the noisy input sequence $\mathbf{x}_t$ (along with the timestep embedding $\mathbf{e}_t$) before being fed into the Transformer. This allows the denoiser's self-attention mechanism to explicitly attend to and leverage information about the potentially inconsistent component, aiding in its robust reconstruction.

(2) Global Context Condition ($\mathbf{c}_{he}$): To maintain overall H-Fact coherence and mitigate both *Intra-fact Inconsistency* and *Cross-fact Association Noise* by leveraging global structural information, we use the CGE's consistency-regularized H-Fact (hyperedge) embedding $\mathbf{e}_f \in \mathbf{E}_H$. A learned projection layer transforms this embedding into the global context condition vector $\mathbf{c}_{he} = \text{Proj}_{he}(\mathbf{e}_f) \in \mathbf{R}^d$. This vector biases the Transformer's self-attention scores, guiding the model towards reconstructions that are consistent with the learned robust global representation of the H-Fact:

$$\text{score}(i, j) = \frac{\mathbf{q}_i^T\mathbf{k}_j}{\sqrt{d_k}} + (\mathbf{W}_q\mathbf{q}_i)^T(\mathbf{W}_c\mathbf{c}_{he}) \quad (6)$$

where $\mathbf{q}_i$ and $\mathbf{k}_j$ are the query and key vectors for positions i and j, d_k is the key dimension, and $\mathbf{W}_q, \mathbf{W}_c$ are learned weight matrices. The second term serves as a global context bias, computing a relevance score between the query at position i and the global H-Fact context $\mathbf{c}_{he}$, which is then broadcast to all keys to uniformly guide attention toward elements consistent with the fact's overall meaning.

By synergistically conditioning the diffusion process on both local uncertainty signals ($\mathbf{c}_{lc}$) derived from component confidence and the global structural context ($\mathbf{c}_{he}$) from the robust H-Fact embedding, the CGD effectively learns to reverse the diffusion process and predict clean H-Fact embeddings $\hat{\mathbf{x}}_0 = [\hat{\mathbf{e}}_1, \ldots, \hat{\mathbf{e}}_L]$, achieving robust handling of the identified HKG noise types.

3.3 Model Training and Inference

HDiff is trained end-to-end by minimizing a composite loss $\mathcal{L}$ (Eq. 7) combining three objectives.

$$\mathcal{L} = \mathcal{L}_{CGE} + \mathcal{L}_{diffusion} + \lambda_{pred}\mathcal{L}_{pred} \quad (7)$$

The first objective, $\mathcal{L}_{CGE}$ (Eq. 3), optimizes the Consistency-Enhanced Global Encoder to produce robust initial embeddings using the combined

Dataset	H-Facts	Triple with Q(%)	Arity	Entities	Relations	Train/Valid/Test
JF17K	100,947	46,320 (45.9%)	2-6	28,645	501	76379 (75.7%) 24.3%
WD50K	236,507	32,167 (13.6%)	2-67	47,155	531	166435 (70.4%) 10.1% 19.5%
WikiPeople	369,866	9,482 (2.6%)	2-7	34,825	178	294439 (79.6%) 10.2% 10.2%

Table 2: Statistics of the hyper-relational datasets

TransE and Barlow Twins objectives. The second is $\mathcal{L}_{diffusion}$, the standard conditional DDPM objective (Ho et al., 2020), which trains the CGD's denoiser ϵ_θ to predict the noise ϵ added to the clean data $\mathbf{x}_0$ at timestep t, conditioned on the noisy state $\mathbf{x}_t$, timestep t, and derived conditions $\mathbf{c}_{lc}, \mathbf{c}_{he}$:

$$\mathcal{L}_{diffusion} = \mathbf{E}_{t\sim U(1,T),\mathbf{x}_0\sim p(\mathbf{x}_0),\epsilon\sim\mathcal{N}(0,\mathbf{I})} \left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t, \mathbf{c}_{lc}, \mathbf{c}_{he})\|^2\right] \quad (8)$$

Finally, $\mathcal{L}_{pred}$ bridges the denoising process with the downstream link prediction task. After estimating the clean H-Fact embedding sequence $\hat{\mathbf{x}}_0$ from the denoiser's output (following the DDPM reverse process), we extract the estimated clean embedding $\hat{\mathbf{e}}_i$ for a component i and map it to vocabulary scores using a linear Prediction Head:

$$\text{Scores}_i = \mathbf{W}_{pred}\hat{\mathbf{e}}_i + \mathbf{b}_{pred} \quad (9)$$

The prediction loss is the cross-entropy between these scores and the ground truth element:

$$\mathcal{L}_{pred} = \mathbf{E}_{\mathbf{x}_0,t,i}\left[-\log \text{Softmax}(\text{Scores}_i)_{\text{true\_element}_i}\right] \quad (10)$$

This objective optimizes the Prediction Head and provides a task-specific signal for fine-tuning both the CGD and CGE. The hyperparameter λ_{pred} balances its contribution in the overall loss.

Inference: For an HKGC query involving a partially observed H-Fact f_{query} with masked element(s), HDiff performs the following inference procedure. First, the trained CGE encodes the known components of f_{query} to derive the conditions $\mathbf{c}_{lc}$ and $\mathbf{c}_{he}$. Then, the conditional reverse diffusion process is performed: starting from a noisy state $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$, the trained conditional denoiser ϵ_θ is iteratively applied for $t = T, T-1, \ldots, 1$ to reconstruct the estimated clean H-Fact embedding sequence $\hat{\mathbf{x}}_0$. Finally, to obtain completion predictions, the reconstructed embedding(s) $\hat{\mathbf{e}}_i$ at the masked position(s) i are extracted from $\hat{\mathbf{x}}_0$ and mapped to vocabulary scores using the linear Prediction Head (Eq. 9). Candidates are then ranked based on their predicted scores. Detailed training procedures and algorithmic steps are provided in Algorithm 1, with a complexity analysis of HDiff in Appendix C.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate on three standard HKG benchmarks: JF17K (Wen et al., 2016b), WikiPeople (Guan et al., 2019), and WD50K (Fatemi et al., 2022), which cover diverse domains and qualifier prevalence. Dataset statistics are in Table 2.

Baselines. We compare HDiff with representative HKGC methods, including translational (m-TransH (Wen et al., 2016b)), compositional/logic-based (NaLP (Guan et al., 2019)), geometric (HINGE (Rosso et al., 2020), sHINGE (Lu et al., 2024)), and recent GNN/Transformer models (StarE (Fatemi et al., 2022), GRAN (Xu et al., 2021), HAHE (Luo et al., 2023)). HAHE serves as a key state-of-the-art attention-based comparison.

Evaluation Metrics. We report standard filtered Mean Reciprocal Rank (MRR) and Hits@K (K=1, 10). Evaluation covers predicting all missing H-Fact components.

LLM Comparison. We conduct qualitative probes using GPT-4o(Achiam et al., 2023) to illustrate the challenges LLMs face in handling HKGC's structural complexity and precise output. Details are in Appendix E.3.

Implementation Details. HDiff is implemented in PyTorch and trained on NVIDIA RTX 4090 GPUs. Key hyperparameters were tuned via validation. Full implementation details and hyperparameter settings are provided in Appendix A.

4.2 Main Results

Table 3 shows that HDiff consistently achieves state-of-the-art (SOTA) entity prediction performance, outperforming all baselines.

HDiff significantly surpasses traditional methods (e.g., m-TransH), which often inadequately handle qualifiers, making them inherently vulnerable to *Intra-fact Inconsistency*. It also shows clear advantages over more sophisticated structural models (e.g., HINGE, StarE, GRAN). While these capture graph structure, their standard scoring or aggregation mechanisms lack explicit consistency enforcement, potentially amplifying *Intra-fact Incon-*

Model	JF17K						WikiPeople						WD50K					
	subject/object			all entities			subject/object			all entities			subject/object			all entities		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
m-TransH (Wen et al., 2016b)	0.206	0.206	0.462	0.102	0.069	0.168	0.063	0.063	0.300	-	-	-	-	-	-	-	-	-
RAE (Zhang et al., 2018)	0.215	0.215	0.466	0.310	0.219	0.504	0.058	0.058	0.306	0.172	0.102	0.320	-	-	-	-	-	-
NaLP (Guan et al., 2019)	0.221	0.165	0.331	0.366	0.290	0.516	0.408	0.331	0.546	0.338	0.272	0.466	-	-	-	0.224	0.158	0.330
NeuInfer (Guan et al., 2020)	0.449	0.361	0.624	0.473	0.397	0.618	0.476	0.415	0.585	0.333	0.259	0.477	0.243	0.176	0.377	0.228	0.162	0.341
HINGE (Rosso et al., 2020)	0.431	0.342	0.611	0.517	0.436	0.675	0.342	0.272	0.463	0.350	0.282	0.467	-	-	-	0.232	0.164	0.343
sHINGE (Lu et al., 2024)	0.458	0.372	0.628	-	-	-	0.478	0.425	0.586	-	-	-	-	-	-	-	-	-
StarE (Fatemi et al., 2022)	0.574	0.496	0.725	0.542	0.454	0.685	0.491	0.398	0.592	0.378	0.265	0.542	0.349	0.271	0.496	-	-	-
Hyper2 (Yan et al., 2021)	0.583	0.500	0.746	-	-	-	0.461	0.391	0.597	-	-	-	-	-	-	-	-	-
HyTransformer (Yu and Yang, 2021)	0.582	0.501	0.742	-	-	-	0.501	0.426	0.634	-	-	-	0.356	0.281	0.498	-	-	-
GRAN (Xu et al., 2021)	0.617	0.539	0.770	0.656	0.582	0.799	0.503	0.438	0.620	0.479	0.410	0.604	-	-	-	0.309	0.240	0.441
HAHE (Luo et al., 2023)	0.623	0.554	0.806	0.668	0.597	0.816	0.509	**0.447**	0.639	0.495	0.420	0.631	0.368	0.291	**0.516**	0.402	**0.327**	**0.546**
HDiff	**0.637**	**0.562**	**0.812**	**0.676**	**0.605**	**0.822**	**0.512**	**0.447**	**0.642**	**0.502**	0.430	**0.639**	**0.373**	**0.292**	0.512	**0.405**	0.325	**0.546**
HDiff w/o MLP	0.627	0.555	0.797	0.669	0.601	0.807	0.506	0.430	0.632	**0.502**	**0.433**	0.631	0.356	0.285	0.498	0.387	0.317	0.528
HDiff w/o TransE	0.620	0.545	0.770	0.661	0.591	0.800	0.485	0.417	0.623	0.478	0.415	0.627	0.354	0.270	0.484	0.375	0.296	0.492
HDiff w/o BT	0.621	0.548	0.769	0.662	0.593	0.799	0.488	0.423	0.625	0.478	0.416	0.627	0.355	0.271	0.484	0.375	0.296	0.495
HDiff w/o LC	0.633	0.559	0.785	0.673	0.604	0.814	0.507	0.424	0.633	0.489	0.424	0.630	0.363	0.288	0.504	0.394	0.322	0.535
HDiff w/o HE	0.622	0.548	0.771	0.663	0.593	0.800	0.488	0.422	0.625	0.479	0.417	0.627	0.356	0.271	0.483	0.375	0.298	0.496

Table 3: Performance Comparison on Hyper-Relational Knowledge Graph Completion. Entity prediction results on JF17K, WikiPeople, and WD50K datasets are presented. The best result in each column is **bolded**, and the second best is underlined. '-' indicates results not reported or available.

sistency during representation learning and propagating distortions caused by *Cross-fact Association Noise*. Notably, HDiff also outperforms recent strong attention-based competitors like HAHE. Although HAHE utilizes attention to weigh components, its mechanism can still be misled by internally conflicting information (*Intra-fact Inconsistency*) without explicit regularization, and like other discriminative models, it struggles to fundamentally correct for noise compared to HDiff's generative reconstruction approach. For instance, on JF17K (subject/object), HDiff's MRR of 0.637 surpasses HAHE's 0.623.

HDiff's superiority stems from its design tailored for noisy HKGs. The CGE provides robust initial embeddings via consistency-enhanced graph learning ($\mathcal{L}_{BT}, \mathcal{L}_{TransE}$). The CGD then employs conditional diffusion, uniquely guided by both local confidence ($\mathbf{c}_{lc}$) and global context ($\mathbf{c}_{he}$), enabling it to reconstruct clean facts generatively. This strategy allows HDiff to handle HKG noise patterns for more accurate completions.

4.3 Ablation Study

We conducted a critical ablation study to assess the distinct contribution of each component within HDiff. By systematically removing modules or loss terms and evaluating performance on clean data (Table 3), we observed consistent performance degradation across metrics, validating our integrated design.

Removing supporting components like the MLP adaptation layer ('w/o MLP') or the TransE loss ('w/o TransE') resulted in relatively minor performance drops. Their roles are more foundational for basic structure and alignment than primary noise handling capabilities.

Ablating core components designed for robustly handling HKG noise led to more significant performance degradation. Specifically, removing the Barlow Twins consistency loss ('w/o BT'), vital for intra-fact consistency, or the Hyperedge Context condition ('w/o HE'), crucial for global coherence, caused notable decreases across metrics and datasets. The Local Confidence condition ('w/o LC') also contributed positively.

Collectively, these results highlight the necessity of each component for achieving robust performance. For a detailed quantitative analysis of their functional roles and critical impact on robustness under noise, please refer to Appendix D.

4.4 Justification of the Diffusion Module

Method	Clean MRR	10% Noise MRR	Rel. Drop
GAT+CL (w/o Diffusion)	0.631	0.487	-22.8%
HDiff (Full Model)	**0.637**	**0.562**	**-11.8%**

Table 4: Impact of the diffusion module on robustness on JF17K. The diffusion process significantly mitigates performance degradation in noisy settings.

To validate the diffusion model's role as a core denoising component, we conducted an ablation study comparing the full HDiff model against a variant excluding the CGD module. As presented in Table 4, the encoder alone offers some robustness to noise; however, the diffusion module significantly enhances performance. Its iterative refinement process reduces performance degradation

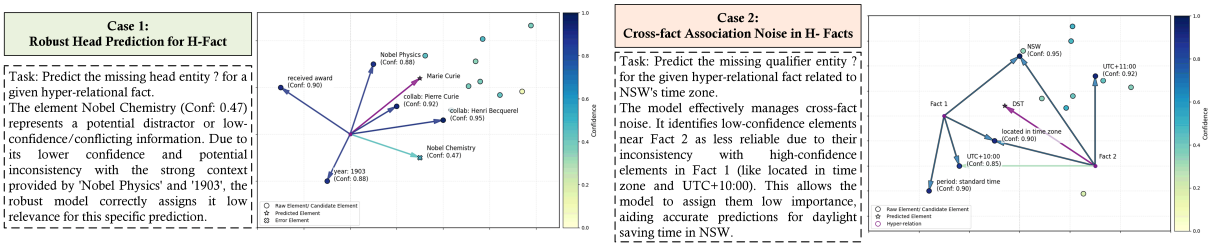

Figure 3: llustrative case studies demonstrating HDiff's handling of noise. Confidence scores (color bar) from $\mathcal{P}_{conf}$ identify potentially inconsistent elements. Case 1 shows overcoming conflicting object information (*Intra-fact Inconsistency*) for head prediction. Case 2 shows resolving conflicting qualifiers/objects for qualifier prediction.

under 10% noise from 22.8% to 11.8%, underscoring its critical role in accurate fact reconstruction in noisy settings.

4.5 Noise Robustness Analysis

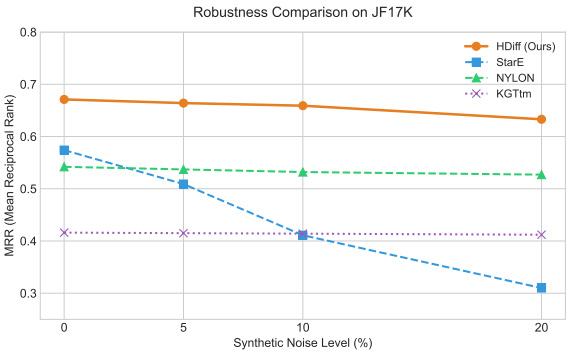

Figure 4: Robustness comparison (MRR) on JF17K against HKGC and robust KGC baselines under increasing synthetic noise.

To evaluate HDiff's robustness, a key motivation as discussed in Section 2, we compare HDiff with both HKGC models (GRAN, NYLON) and a traditional robust KGC baseline, KGTtm (Jia et al., 2019), on the JF17K training set with synthetic noise from 2%-20%. KGTtm was chosen as it can be fairly applied to score H-Facts directly, whereas other methods like CKRL would require substantial, non-trivial modifications to handle hyper-relational structures. Figure 4 shows that while all models degrade with increasing noise, HDiff consistently maintains the highest MRR. Notably, HDiff's performance degrades more gracefully than other methods, especially at higher noise levels. While NYLON also shows robustness, HDiff retains a clear advantage across the noise spectrum. Crucially, unlike prior methods requiring external noise labels or complex pre-processing, HDiff's superior resilience stems from its intrinsic, learnable noise handling components. This superior resilience validates HDiff's design; its conditional diffusion process, leveraging CGE's consistency regularization ($\mathcal{L}_{BT}$) and CGD's dual conditioning ($\mathbf{c}_{lc}, \mathbf{c}_{he}$), effectively mitigates noise impact during reconstruction. A detailed component-wise analysis of how HDiff achieves this robustness under noise through ablation is provided in Appendix E.

4.6 Case Study

Figure 3 illustrates HDiff's robustness via scenarios: confidence estimation ($\mathcal{P}_{conf}$) identifies unreliable components, guiding CGD through $\mathbf{c}_{lc}$ and leveraging global hyperedge features ($\mathbf{c}_{he}$).

Case 1 (Intra-fact Inconsistency): Predicting head entity 'Marie Curie' for Nobel prize in '1903'. Context contains conflicting objects: 'Nobel Physics' (relevant, high confidence) and 'Nobel Chemistry' (distractor, 0.47 confidence). This inconsistency is characteristic of *Intra-fact Inconsistency*. $\mathcal{P}_{conf}$ identifies 'Nobel Chemistry' as low confidence, informing $\mathbf{c}_{lc}$. Global context $\mathbf{c}_{he}$ reflects coherent structure (Physics, 1903, collaborators). CGD integrates both, down-weighting inconsistent 'Chemistry' and correctly predicting 'Marie Curie'.

Case 2 (Intra-fact Inconsistency): Predicting qualifier value 'DST' for NSW time zone, given object 'UTC+10:00' (Standard Time). This object conflicts with the typical DST offset (UTC+11:00). This represents another form of *Intra-fact Inconsistency*. $\mathcal{P}_{conf}$ assigns low confidence to 'UTC+10:00', influencing $\mathbf{c}_{lc}$. Global context $\mathbf{c}_{he}$ captures broader association (NSW, DST, UTC+11:00). Guided by local uncertainty ($\mathbf{c}_{lc}$) and global knowledge ($\mathbf{c}_{he}$), CGD correctly predicts 'DST'.

These cases demonstrate how HDiff's internal confidence assessment flags local inconsistencies, while global context ensures structural coherence,

enabling robust predictions even with noisy or conflicting H-facts.

5 Conclusion

We introduced **HDiff**, a novel conditional diffusion framework addressing *Intra-fact Inconsistency* and *Cross-fact Association Noise* for robust Hyper-relational Knowledge Graph Completion (HKGC). By integrating a **Consistency-Enhanced Global Encoder (CGE)** with contrastive learning and a **Context-Guided Denoiser (CGD)** employing unique dual conditioning, HDiff facilitates robust reconstruction of clean H-Facts from noisy inputs. Experiments demonstrated state-of-the-art performance and superior noise robustness, with ablations validating core component contributions. HDiff represents a notable advancement towards reliable HKGC in complex and noisy environments. Future work can explore efficiency improvements and broader hyper-relational reasoning tasks.

Limitations

Despite its strong performance, HDiff has several limitations. Firstly, integrating Transformer architectures and the iterative denoising process inherent in diffusion models results in a higher computational cost compared to simpler baseline models. Training and inference require significant computational resources and time, which could pose challenges for deployment on resource-constrained platforms or extremely large-scale knowledge graphs. Secondly, like many neural models, HDiff's performance can be sensitive to hyperparameter choices, requiring careful tuning. Finally, our robustness analysis was limited to synthetic noise levels between 2% and 20%. While this range is relevant for evaluating HDiff's intrinsic learned denoising in practical scenarios (Section 2.2, Appendix D), the model's behavior under significantly higher or qualitatively different noise distributions remains unexplored. Investigating HDiff's robustness across a wider and more diverse noise spectrum is a crucial direction for future work.

Ethics Statement

This study complies with the ACL Ethics Policy.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (2021YFC3300602).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems (NeurIPS)*, volume 26.

Agustín Borrego, Daniel Ayala, Inma Hernández, Carlos R. Rivero, and David Ruiz. 2019. Generating rules to filter candidate triples for their correctness checking by knowledge graph completion techniques. In *Proceedings of the 10th International Conference on Knowledge Capture*, K-CAP '19, page 115–122, New York, NY, USA. Association for Computing Machinery.

Preetha Datta, Fedor Vitiugin, Anastasiia Chizhikova, and Nitin Sawhney. 2024. Construction of hyper-relational knowledge graphs using pre-trained large language models. *arXiv preprint arXiv:2403.11786*.

Na Dong, Natthawut Kertkeidkachorn, Xin Liu, and Kiyoaki Shirai. 2025. Refining noisy knowledge graph with large language models. In *Proceedings of the Workshop on Generative AI and Knowledge Graphs (GenAIK)*, pages 78–86, Abu Dhabi, UAE. International Committee on Computational Linguistics.

Bahare Fatemi, Ajay Pavan, Harsh Gupta, Megha Jaggi, and Foaad Havasi. 2022. Message passing for hyper-relational knowledge graphs. In *International Conference on Learning Representations (ICLR)*.

Yifan Feng, Chengwu Yang, Xingliang Hou, Shaoyi Du, Shihui Ying, Zongze Wu, and Yue Gao. 2024. Beyond graphs: Can large language models comprehend hypergraphs? *arXiv preprint arXiv:2410.10083*.

Saiping Guan, Xiaolong Jin, Jiafeng Guo, Yuanzhuo Wang, and Xueqi Cheng. 2020. Neuinfer: Knowledge inference on n-ary facts. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Saiping Guan, Xiaolong Jin, Yuanzhuo Wang, and Xueqi Cheng. 2019. Link prediction on n-ary relational data. In *Proceedings of the 2019 World Wide Web Conference (WWW)*, pages 583–593. ArXiv: https://arxiv.org/abs/1708.01411.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems 33 (NeurIPS)*, pages 6840–6851. ArXiv: https://arxiv.org/abs/2006.11239.

Aidan Hogan, Stefan Bloehdorn, Philipp Cimiano, Raúl García-Castro, Claudia d'Amato, Antoine Zimmermann, Steffen Staab, and Volker Tresp. 2021. Knowledge graphs. *ACM Computing Surveys (CSUR)*, 54(4):1–37.

Yan Hong, Chenyang Bu, and Tingting Jiang. 2020. Rule-enhanced noisy knowledge graph embedding via low-quality error detection. In *2020 IEEE International Conference on Knowledge Graph (ICKG)*, pages 544–551.

Zhi-Yu Hong and Zongmin Ma. 2023. Inconsistency detection in knowledge graph with entity and path semantics. *J. Inf. Sci. Eng.*, 39(2):291–303.

Shengbin Jia, Yang Xiang, Xiaojun Chen, and Kun Wang. 2019. Triple trustworthiness measurement for knowledge graph. In *The World Wide Web Conference*, pages 2865–2871.

Yangqin Jiang, Yuhao Yang, Lianghao Xia, and Chao Huang. 2024. Diffkg: Knowledge graph diffusion model for recommendation. In *Proceedings of the 17th ACM international conference on web search and data mining*, pages 313–321.

Yu Liu, Quanming Yao, and Yong Li. 2020. Generalizing tensor decomposition for n-ary relational knowledge bases. In *Proceedings of The Web Conference 2020*, WWW '20, page 1104–1114, New York, NY, USA. Association for Computing Machinery.

Xiao Long, Liansheng Zhuang, Aodi Li, Jiuchang Wei, Houqiang Li, and Shafei Wang. 2024. Kgdm: A diffusion model to capture multiple relation semantics for knowledge graph embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 8850–8858.

Yuhuan Lu, Dingqi Yang, Pengyang Wang, Paolo Rosso, and Philippe Cudre-Mauroux. 2024. Schema-aware hyper-relational knowledge graph embeddings for link prediction. *IEEE Transactions on Knowledge and Data Engineering*, 36(6):2614–2628.

Haoran Luo, Haihong E, Yuhao Yang, Yikai Guo, Mingzhi Sun, Tianyu Yao, Zichen Tang, Kaiyang Wan, Meina Song, and Wei Lin. 2023. HAHE: hierarchical attention for hyper-relational knowledge graphs in global and local level. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 8095–8107. Association for Computational Linguistics.

Paolo Rosso, Dingqi Yang, and Philippe Cudré-Mauroux. 2020. Beyond the triples: Hyper-relational knowledge graph embedding for link prediction. In *Proceedings of The Web Conference 2020 (WWW)*, pages 2351–2361. ArXiv: https://arxiv.org/abs/2001.09933.

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

Xiaojuan Tang, Song-Chun Zhu, Yitao Liang, and Muhan Zhang. 2024. RulE: Knowledge graph reasoning with rule embedding. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4316–4335, Bangkok, Thailand. Association for Computational Linguistics.

Victor M. Tenorio, Samuel Rey, and Antonio G. Marques. 2023. Robust graph neural network based on graph denoising. In *2023 57th Asilomar Conference on Signals, Systems, and Computers*, pages 578–582.

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

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Yanbin Wei, Qiushi Huang, James T Kwok, and Yu Zhang. 2024. Kicgpt: Large language model with knowledge in context for knowledge graph completion. *arXiv preprint arXiv:2402.02389*.

Jiacheng Wen, Jicong Chen, Zhan Li, Chengzheng Zhang, and Weinan Chen. 2016a. On the representation and embedding of knowledge bases with n-ary relational facts. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1313–1319.

Jianfeng Wen, Jianxin Li, Yongyi Mao, Shini Chen, and Richong Zhang. 2016b. On the representation and embedding of knowledge bases beyond binary relations. *arXiv preprint arXiv:1604.08642*.

Wenbin Xiong, Zhan Liu, Yihan Yang, Ge Yu, Ge Zhang, and Tian Song. 2023. A survey on hyper-relational knowledge graphs: From data models to applications. *arXiv preprint arXiv:2301.03869*.

Mengfei Xu, Lei Wu, and Jian Li. 2021. Gran: Graph relational attention network for multi-relational reasoning. In *Proceedings of the AKBC 2021 Workshop on Knowledge Base Construction, Reasoning, and Mining (AKBC)*.

Qi Yan, Jiaxin Fan, Mohan Li, Guanqun Qu, and Yang Xiao. 2022. A survey on knowledge graph embedding. In *7th IEEE International Conference on Data Science in Cyberspace, DSC 2022, Guilin, China, July 11-13, 2022*, pages 576–583. IEEE.

Shiyao Yan, Zequn Zhang, Xian Sun, Guangluan Xu, Li Jin, and Shuchao Li. 2021. $hyper^2$: Hyperbolic poincare embedding for hyper-relational link prediction. *ArXiv*, abs/2104.09871.

Liang Yao, Jiazhen Peng, Chengsheng Mao, and Yuan Luo. 2025. Exploring large language models for knowledge graph completion. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Donghan Yu and Yiming Yang. 2021. Improving hyper-relational knowledge graph completion. *ArXiv*, abs/2104.08167.

Weijian Yu, Jie Yang, and Dingqi Yang. 2024. Robust link prediction over noisy hyper-relational knowledge graphs via active learning. *Proceedings of the ACM on Web Conference 2024*.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. 2021. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning (ICML)*, pages 12263–12274. PMLR. ArXiv: https://arxiv.org/abs/2103.03230.

Richong Zhang, Junpeng Li, Jiajie Mei, and Yongyi Mao. 2018. Scalable instance reconstruction in knowledge bases via relatedness affiliated embedding. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1185–1194, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

A Hyperparameter Settings

We performed hyperparameter tuning for our proposed HDiff model using a grid search strategy over the parameter ranges specified in Table 5. The selection was based on optimizing the Hits@1 metric for the 'all entities' prediction task on the respective validation set of each dataset (JF17K, WikiPeople, WD50K). The performance for each combination was typically evaluated after a fixed number of initial training epochs (60 epochs) to efficiently explore the search space.

The key hyperparameters tuned include: the main embedding dimension (d); the number of GAT layers (L_{GAT}) in the CGE; the dimension used for noise/timestep embeddings; the number of diffusion timesteps (T); the number of Transformer layers (L_{TF}) in the CGD; the training batch size; the learning rate; and the weights for the auxiliary losses in the CGE, specifically the TransE loss weight (λ_{TransE}) and the Barlow Twins contrastive loss weight (λ_{BT}).

Table 5 lists the search space explored for each parameter and indicates the final optimal value selected for each dataset, which is underlined.

B Results of Relation Prediction

As demonstrated in Table 6, previous models for HKG embedding have yielded excellent outcomes in relation prediction, with certain metrics even attaining a success rate of 99%.

C Training Procedure and Complexity Analysis

This appendix details the training algorithm for the HDiff framework described in Section 3 and provides an analysis of its computational complexity.

C.1 Training Algorithm

Algorithm 1 outlines the end-to-end training process for HDiff, integrating the optimization of both the CGE and CGD modules based on the combined loss function defined in Eq. 7.

C.2 Complexity Analysis

The computational cost per training iteration (batch) is primarily driven by the forward passes through the CGE's GAT encoder and the CGD's Transformer denoiser (ϵ_θ). Let B be the batch size, N the number of entities, M the number of H-Facts in the batch, E the number of edges/relations processed by the GNN, $\bar{L}$ the average H-Fact sequence length (number of components), d the embedding dimension, and L_{TF} the number of layers in the Transformer denoiser.

The GNN cost depends on its specific architecture; for GAT, it is roughly related to the number of edges processed. However, the most computationally intensive component is often the Transformer denoiser due to its self-attention mechanism, which scales quadratically with the sequence length $\bar{L}$.

Ignoring the GNN cost (which can vary) and focusing on the dominant Transformer part, the time complexity per batch for the CGD forward pass is approximately:

$$O(B \cdot L_{TF} \cdot \bar{L}^2 \cdot d) \tag{11}$$

This highlights that the model's runtime is particularly sensitive to the length of the H-Fact representations ($\bar{L}$), corresponding to the number of components (core triple + qualifiers) in a hyper-relational fact. Efficient implementation is crucial, especially for HKGs where facts may contain numerous qualifiers.

D Additional Ablation Studies and Model Justifications

D.1 Encoder Comparison

To justify our choice of the Graph Attention Network (GAT) as the primary encoder in the CGE module, we conducted a comparative analysis against other commonly used Graph Neural Networks: Graph Convolutional Network (GCN) and

Hyperparameter	JF17K	WikiPeople	WD50K
Embedding Dimension (d)	{128, 256, 512}	{128, 256, 512}	{128, 256, 512}
GAT Layers (L_{GAT})	{1, 2, 3, 4}	{1, 2, 3, 4}	{1, 2, 3, 4}
Noise/Timestep Emb. Dim.	{12, 16, 20, 24}	{12, 16, 20, 24}	{12, 16, 20, 24}
Diffusion Timesteps (T)	{10, 13, 15, 20}	{10, 13, 15, 20}	{10, 13, 15, 20}
Transformer Layers (L_{TF})	{2, 4, 8, 12}	{2, 4, 8, 12}	{2, 4, 8, 12, 16}
TransE Loss Weight (λ_{TransE})	{0.1, 0.5, 1.0}	{0.1, 0.5, 1.0}	{0.1, 0.5, 1.0}
Barlow Twins Loss Weight (λ_{BT})	{0.1, 0.5, 1.0, 2.0}	{0.1, 0.5, 1.0, 2.0}	{0.1, 0.5, 1.0, 2.0}
Batch Size	{64, 256, 512}	{64, 256, 512}	{64, 256, 512}
Learning Rate	{0.0001, 0.0005, 0.001}	{0.0001, 0.0005, 0.001}	{0.00005, 0.0001, 0.0005}

Table 5: Hyperparameter search space and selected optimal values (underlined) for HDiff on JF17K, WikiPeople, and WD50K.

Model	JF17K						WikiPeople						WD50K					
	subject/object			all entities			subject/object			all entities			subject/object			all entities		
	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10	MRR	H@1	H@10
m-TransH (Wen et al., 2016b)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
RAE	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
NaLP (Guan et al., 2019)	0.639	0.547	0.822	0.825	0.762	0.927	0.482	0.32	0.482	0.735	0.595	0.938	-	-	-	-	-	-
NeuInfer (Guan et al., 2020)	0.936	0.901	0.989	-	-	-	0.95	0.915	0.997	-	-	-	-	-	-	-	-	-
HINGE (Rosso et al., 2020)	-	-	-	0.861	0.832	0.91	-	-	-	0.765	0.686	0.9	-	-	-	-	-	-
sHINGE (Lu et al., 2024)	0.966	0.943	0.996	-	-	-	0.950	0.917	0.997	-	-	-	-	-	-	-	-	-
StarE (Fatemi et al., 2022)	-	-	-	0.901	0.884	0.963	-	-	-	0.378	0.265	0.542	-	-	-	-	-	-
Hyper2 (Yan et al., 2021)	0.95	0.933	0.976	-	-	-	0.947	0.914	0.987	-	-	-	-	-	-	-	-	-
HyTransformer (Yu and Yang, 2021)	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GRAN (Xu et al., 2021)	0.992	0.988	0.988	0.996	0.993	**0.999**	0.957	0.942	0.976	0.96	0.946	0.977	-	-	-	-	-	-
HAHE (Luo et al., 2023)	0.993	0.989	**0.998**	0.996	0.994	**0.999**	0.957	0.941	0.978	0.958	0.942	0.978	0.916	0.885	0.964	0.927	0.900	0.969
HDiff	**0.995**	**0.993**	**0.998**	**0.997**	**0.996**	**0.999**	0.973	**0.958**	0.991	0.973	0.959	0.992	**0.950**	0.924	0.985	0.956	**0.934**	**0.987**
HDiff w/o MLP	0.994	0.992	**0.998**	0.996	0.994	**0.999**	0.972	**0.958**	0.990	0.972	0.958	0.990	**0.950**	**0.925**	0.985	0.956	**0.934**	**0.987**
HDiff w/o TransE	0.994	0.991	0.997	0.995	0.993	0.998	0.972	0.957	0.990	0.972	0.957	0.990	0.949	0.923	0.985	0.956	0.933	**0.987**
HDiff w/o BT	0.994	0.991	**0.998**	0.995	0.992	0.998	0.972	**0.958**	0.991	0.972	0.956	0.990	0.949	0.923	**0.986**	**0.957**	**0.934**	**0.987**
HDiff w/o LC	**0.995**	0.992	**0.998**	**0.997**	**0.995**	**0.999**	**0.976**	0.961	0.994	**0.976**	**0.962**	**0.994**	0.947	0.921	0.985	0.954	0.931	0.986
HDiff w/o HE	**0.995**	0.992	**0.998**	0.996	0.994	**0.999**	0.975	0.961	**0.994**	**0.976**	0.961	**0.994**	0.946	0.921	0.985	0.954	0.930	0.986

Table 6: Performance Comparison on Hyper-Relational Knowledge Graph Completion – **Relation Prediction Task**. Results (MRR, Hits@1, Hits@10) on JF17K, WikiPeople, and WD50K datasets. The best result in each column is **bolded**, and the second best is underlined. '-' indicates results not reported or available.

Relational Graph Convolutional Network (R-GCN). The experiment was performed on the JF17K dataset under both clean and 10% noisy conditions.

As shown in Table 7, while all encoders perform reasonably on clean data, GAT demonstrates significantly superior performance in the noisy setting. Its ability to dynamically assign attention weights to different components in an H-Fact allows it to better mitigate the impact of noisy or irrelevant qualifiers, leading to a more robust representation. This empirical result validates GAT as the most suitable encoder for our robust HKGC framework.

Encoder	**Parameters**	**Clean**	**10% Noise**
GCN	21 M	0.614	0.437
R-GCN	23 M	0.621	0.441
GAT (Ours)	22 M	**0.637**	**0.562**

Table 7: MRR Performance of GNN Encoders on JF17K.

D.2 Parameter Sensitivity Analysis

We analyzed HDiff's sensitivity to key hyperparameters (Figure 5). Performance generally peaks with diffusion steps T around 10-20 (Fig. 5a), balancing refinement and efficiency. The optimal diffusion dimension d is often observed around 16-24 (Fig. 5b). Deeper models benefit performance, with Transformer layers L_{TF} typically improving up to 8-12 before saturation (Fig. 5c), while GAT layers L_{GAT} show optimal performance around 2 or 3 (Fig. 5d). Overall, HDiff demonstrates reasonable stability across typical hyperparameter ranges, indicating it is not overly sensitive to tuning.

E Detailed Robustness Analysis and Noise Methodology

This appendix details the methodology behind our noise analysis and presents an in-depth evaluation of HDiff's robustness. First, we provide an empirical analysis of natural noise in the benchmark datasets and detail our synthetic noise generation strategy. Second, we present the full ablation study

Algorithm 1 HDiff Training Procedure

Input: Training HKG $\mathcal{H}$, batch size B, learning rate η, loss weights $\lambda_{TransE}, \lambda_{BT}$, diffusion steps T, total training steps N_{steps}.

Output: Trained HDiff parameters $\Theta = \{\theta_{CGE}, \theta_{CGD}\}$.

for $step = 1$ to N_{steps} **do**
  Sample a mini-batch of B H-Facts $\{f_i\}_{i=1}^{B}$ from $\mathcal{H}$.
  // CGE Module Computations
  Obtain intermediate GNN embeddings (using g_ω) for entities $\mathbf{Z}_{\mathcal{E},i}$ and facts $\mathbf{Z}_{\mathcal{H},i}$ in the batch.
  Compute core triple embeddings $\mathbf{z}_{sro,i}$ from $\mathbf{Z}_{\mathcal{E},i}$.
  Calculate $\mathcal{L}_{TransE}$ using $\mathbf{Z}_{\mathcal{E},i}$ (Eq. 2).
  Calculate $\mathcal{L}_{BT}$ using $\mathbf{Z}_{\mathcal{H},i}$ and $\mathbf{z}_{sro,i}$ (Eq. 1).
  Calculate CGE loss: $\mathcal{L}_{CGE} = \lambda_{TransE}\mathcal{L}_{TransE} + \lambda_{BT}\mathcal{L}_{BT}$.
  Obtain final adapted embeddings and $\mathbf{H}_i$ using MLP adapter g_ψ from $\mathbf{Z}_{\mathcal{E},i}, \mathbf{Z}_{\mathcal{H},i}$.
  // CGD Module Computations (Diffusion)
  Construct initial clean sequences $\{\mathbf{x}_{0,i}\}_{i=1}^{B}$ using adapted embeddings $\mathbf{E}_i$.
  Calculate min confidence conditions $\{\mathbf{c}_{lc,i}\}_{i=1}^{B}$ using $\mathcal{P}_{conf}$ on $\{\mathbf{x}_{0,i}\}$.
  Retrieve and project global hyperedge conditions $\{\mathbf{c}_{he,i}\}_{i=1}^{B}$ from $\mathbf{H}_i$.
  Sample timesteps $\{t_i\}_{i=1}^{B} \sim \text{Uniform}(1, T)$.
  Sample noise $\{\boldsymbol{\epsilon}_i\}_{i=1}^{B} \sim \mathcal{N}(0, \mathbf{I})$.
  Compute noisy sequences $\mathbf{x}_{t_i,i} = \sqrt{\bar{\alpha}_{t_i}}\mathbf{x}_{0,i} + \sqrt{1-\bar{\alpha}_{t_i}}\boldsymbol{\epsilon}_i$.
  Predict noise using the denoiser: $\hat{\boldsymbol{\epsilon}}_i = \boldsymbol{\epsilon}_\theta(\mathbf{x}_{t_i,i}, t_i, \mathbf{c}_{lc,i}, \mathbf{c}_{he,i})$.
  Compute diffusion loss: $\mathcal{L}_{diffusion} = \frac{1}{B}\sum_{i=1}^{B} ||\boldsymbol{\epsilon}_i - \hat{\boldsymbol{\epsilon}}_i||^2$.
  // Combined Loss and Parameter Update
  Calculate the total primary training loss: $\mathcal{L} = \mathcal{L}_{diffusion} + \mathcal{L}_{CGE}$.
  Compute gradient w.r.t all parameters: $\nabla_\Theta \mathcal{L}$.
  Update parameters using optimizer: $\Theta \leftarrow \text{OptimizerUpdate}(\Theta, \nabla_\Theta \mathcal{L}, \eta)$.
end for

results under these noisy conditions to validate the contribution of each model component.

E.1 Noise Analysis and Generation Methodology

E.1.1 Empirical Analysis of Natural Noise

To ground our study in realistic conditions, we conducted an empirical analysis to quantify the natural noise present in the benchmark datasets. For each of the three datasets, we randomly sampled 1,000 H-Facts. These 3,000 samples were then manually annotated by three graduate student annotators in a double-blind setup to identify instances of Intra-fact Inconsistency and Cross-fact Association Noise. The inter-annotator agreement was high, with a Cohen's Kappa coefficient of 0.81. The results, summarized in Table 8, confirm that both noise patterns are prevalent, with a total natural noise rate ranging from 4.1% to 7.5%.

Dataset	Intra-fact (%)	Cross-fact (%)	Total Natural Noise (%)
JF17K	3.7	1.5	**5.2**
WD50K	5.4	2.1	**7.5**
WikiPeople	2.9	1.2	**4.1**

Table 8: Distribution of naturally occurring noise types identified via manual annotation.

E.1.2 Synthetic Noise Generation Strategy

To evaluate model robustness in a controlled manner, we designed a synthetic noise injection strategy that simulates the observed natural noise patterns. The process is detailed in Algorithm 2. The overall noise rate ρ is explicitly allocated to two types: Intra-fact corruption (with probability $p_{\text{intra}} = 0.7$), which simulates *Intra-fact Inconsistency* by randomly replacing elements within a single H-Fact; and Cross-fact corruption (with probability $p_{\text{cross}} = 0.3$), which simulates *Cross-fact Association* by swapping elements between two H-Facts of the same schema. Our synthetic noise distributions align closely with the natural noise ratios found in Table 8. This setup allows for a rigorous evaluation of a model's intrinsic denoising capabilities.

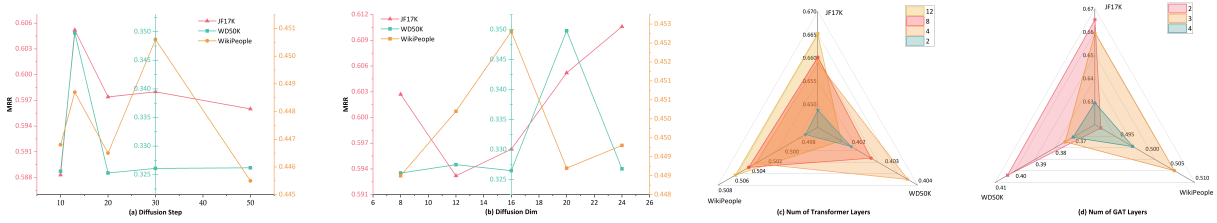

Figure 5: Parameter sensitivity analysis (MRR) on JF17K, WD50K, and WikiPeople for: (a) Diffusion Steps (T), (b) Diffusion Dimension (d), (c) Transformer Layers (L_{TF}), (d) GAT Layers (L_{GAT}).

Algorithm 2 Synthetic Noise Injection

Input: Hyper-relational KG $\mathcal{H}$; overall noise rate ρ; allocation $(p_{\text{intra}}, p_{\text{cross}})$; max corrupt $k_{\max}$.
Output: Corrupted set $\widetilde{\mathcal{H}}$ $\widetilde{\mathcal{H}} \leftarrow \emptyset$.

for each $f = (s, r, o, \{(k_i, v_i)\}_{i=1}^{n}) \in \mathcal{H}$ **do**
  Sample $u \sim \mathcal{U}(0, 1)$.
  if $u \geq \rho$ **then**
    Add f to $\widetilde{\mathcal{H}}$; **continue**.
  end if
  Sample $m \sim \text{UniformInt}[1, k_{\max}]$.
  Sample $z \sim \text{Bernoulli}(p_{\text{intra}})$.
  if $z = 1$ **then**
    // Intra-fact corruption
    Replace m elements in f with random tokens of the same type.
  else
    // Cross-fact corruption
    Select another fact f' with the same schema.
    Swap m aligned elements between f and f'.
  end if
  Add modified fact(s) to $\widetilde{\mathcal{H}}$.
end for
return $\widetilde{\mathcal{H}}$

E.1.3 Noise Sensitivity Analysis

To confirm that HDiff's robustness is not dependent on a specific noise mixture, we conducted a sensitivity analysis on JF17K with 10% noise, testing its performance against different noise types exclusively. As shown in Table 9, HDiff significantly outperforms the strong baseline StarE in all scenarios, with the greatest improvement seen in the mixed-noise setting that mirrors real-world conditions. This demonstrates the versatility of our model's denoising mechanism.

Noise Scenario	StarE	HDiff (Ours)	Improvement
Intra-fact Only	0.418	**0.539**	+28.9%
Cross-fact Only	0.401	**0.524**	+30.7%
Mixed (Original)	0.411	**0.562**	+36.7%

Table 9: Noise sensitivity analysis on JF17K (MRR, 10% noise). HDiff demonstrates robust performance across different noise compositions.

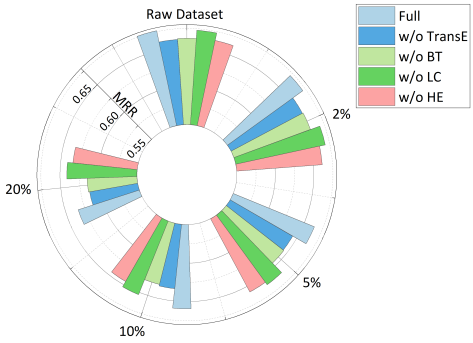

Figure 6: Robustness comparison (MRR) of HDiff and its ablated variants on JF17K with increasing synthetic noise levels (Raw/0%, 2%, 5%, 10%, 20%).

E.2 Robustness Ablation Analysis

We analyze the ablation results from two perspectives: performance on clean data (Table 3) and robustness under synthetic noise (Figure 6).

Analysis on Clean Data Performance. Table 3 in the main paper presents the performance of ablated HDiff variants on clean training/validation/test sets. While Section 4.3 provides a summary, here we offer a more detailed breakdown of each component's impact:

Supporting Components (MLP, TransE Loss): As noted, removing the MLP adaptation layer ('w/o MLP'), which aligns embedding spaces from different encoders, results in minor performance drops (e.g., JF17K all entities MRR: 0.676 $\rightarrow$ 0.669, WikiPeople all entities MRR: 0.509 $\rightarrow$ 0.502). This confirms its role in facilitating representation integration but shows it's not the primary performance bottleneck or noise handler. Similarly, ablating

the TransE-based loss ('w/o TransE'), intended for basic (s, r, o) structure capture, causes slight decreases (e.g., JF17K all entities MRR: 0.676 $\rightarrow$ 0.661, WD50K all entities MRR: 0.405 $\rightarrow$ 0.375). This indicates its foundational role in providing basic structural inductive bias, but its direct contribution to handling complex hyper-relational noise is limited.

Consistency Regularization ($\mathcal{L}_{BT}$): Ablating the Barlow Twins consistency loss ('w/o BT') causes a more significant performance drop (e.g., JF17K all entities MRR: 0.676 $\rightarrow$ 0.662, WD50K all entities MRR: 0.405 $\rightarrow$ 0.375). This loss is vital for the CGE module to learn intra-fact semantic consistency by making representations from different views of the same H-fact consistent. This process creates more robust and less ambiguous initial representations, crucial even on data nominally considered "clean" but potentially containing subtle inconsistencies.

Context-Guided Denoiser (CGD) Conditions (LC, HE): Within CGD, both conditions are important. Removing the Local Confidence condition ('w/o LC'), which guides the denoiser using uncertainty scores ($\mathbf{c}_{lc}$) for individual elements, reduces performance (e.g., WD50K all entities MRR: 0.405 $\rightarrow$ 0.394, JF17K all entities MRR: 0.676 $\rightarrow$ 0.673). This confirms the utility of fine-grained guidance for targeting specific noisy parts. Ablating the global Hyperedge Context condition ('w/o HE'), which uses CGE's robust overall representation ($\mathbf{c}_{he}$) for global coherence guidance, leads to the largest performance drop on clean data (e.g., JF17K all entities MRR: 0.676 $\rightarrow$ 0.663, WD50K all entities MRR: 0.405 $\rightarrow$ 0.375). This emphasizes that maintaining global fact consistency during denoising is paramount for accurate prediction.

Robustness Analysis under Synthetic Noise. Figure 6 visualizes the MRR of the full HDiff model and key ablated variants on JF17K under increasing levels of synthetic noise. This analysis critically demonstrates which components are essential for robustness.

As noise levels rise, all models degrade, but the full HDiff model maintains the highest performance. Ablation under noise reveals:

Impact of $\mathcal{L}_{BT}$ on Robustness: The 'w/o BT' variant shows a notably steeper performance decline curve under noise compared to the full model. This strongly validates that $\mathcal{L}_{BT}$'s role in enforcing intra-fact consistency within CGE is a primary mechanism for building initial representations resilient to *Intra-fact Inconsistency Noise*.

Impact of CGD Conditions on Robustness: Removing the Local Confidence condition ('w/o LC') also accelerates performance degradation under noise, though typically less severely than 'w/o BT' or 'w/o HE'. This confirms that using local uncertainty signals $\mathbf{c}_{lc}$ allows CGD to more effectively pinpoint and correct noisy elements, contributing robustness particularly against localized *Intra-fact Inconsistency*. Ablating the global Hyperedge Context condition ('w/o HE') leads to the most drastic performance drop under noise, showing the steepest decline. This is because $\mathbf{c}_{he}$ guides the denoising process towards overall coherent states, directly counteracting noise that disrupts global fact meaning or cross-element associations (*Intra-fact* and potentially *Cross-fact Association Noise*). Without this global guidance, the denoiser struggles to reconstruct a consistent and meaningful hyper-fact from corrupted input.

Supporting Components' Robustness: Consistent with their minor impact on clean performance, the 'w/o MLP' and 'w/o TransE' variants show less severe degradation under noise compared to the other ablations. This confirms their roles are more about foundational structure and representation alignment rather than direct noise mitigation.

In summary, the ablation results on both clean and noisy data, detailed herein, underscore the synergistic importance of HDiff's components. The CGE's consistency learning ($\mathcal{L}_{BT}$) builds resilient initial representations, while the CGD's dual conditioning ($\mathbf{c}_{lc}, \mathbf{c}_{he}$) provides targeted and globally coherent guidance for effective denoising. This combination is crucial for both high performance and robustly handling the complex, structured noise inherent in HKGs.

E.3 Illustrative Examples and Empirical Observations

F Considerations on Large Language Models for Hyper-Relational Knowledge Graph Completion

While Large Language Models (LLMs) have demonstrated remarkable capabilities across numerous natural language processing tasks, including some promising explorations in traditional Knowledge Graph Completion (KGC) by leveraging textual contexts and entity descriptions (Yao et al., 2025; Wei et al., 2024), their direct and effective application to hyper-relational knowledge graph

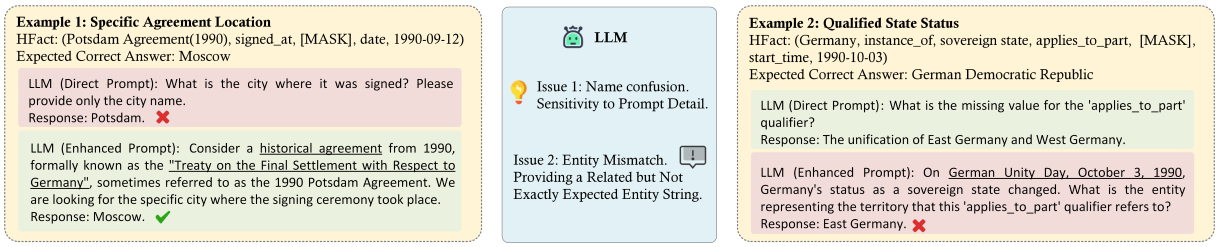

Figure 7: Illustrative examples of hyper-relational facts and sample prompts for LLM-based knowledge graph completion.

completion (HKGC) presents significant, distinct challenges. Hyper-relational knowledge graphs (HKGs) introduce complex, high-order structures involving multiple entities per relation instance, alongside attribute-value pairs modifying relation instances. This inherent structural complexity and the typically higher sparsity compared to traditional KGs make HKGC a different problem space. Although some recent work explores using LLMs for HKG construction from text (Datta et al., 2024) or evaluating their understanding of hypergraphs (Feng et al., 2024), directly employing LLMs for the completion task—predicting missing entities or attributes within existing, complex hyper-relational facts—is not yet a widely established or readily superior paradigm.

To provide empirical context for the challenges discussed and to illustrate practical difficulties general-purpose LLMs may face in HKGC, we conducted small-scale probes using GPT-4o on selected hyper-relational completion tasks from our dataset.

Case Study 1: Specific Agreement Location. Figure 7 (left) shows the task of predicting the signing location for the "Potsdam Agreement (1990)". Probes using both direct and more context-rich prompts revealed that, despite access to general knowledge, the LLM showed susceptibility to name ambiguity inherent in the fact's subject ("Potsdam"). Extracting the precise qualified location associated with this specific historical instance (Moscow), which is stored as a specific attribute value in the HKG, proved challenging for the LLM without explicit structural guidance, often being misled by the subject's name.

Case Study 2: Qualified State Status. Figure 7 (right) test the LLM on completing a fact about Germany's status in a specific context, indicated by the 'applies to part' qualifier. Responses, even with enhanced prompts, frequently exhibited difficulty accurately interpreting the complex qualifier's semantic role within the hyper-relation. The LLM often provided general interpretive text or related but incorrect entities, instead of generating the required specific entity 'German Democratic Republic' fitting the qualifier's role.

These examples empirically demonstrate the difficulty general-purpose LLMs can have in disentangling precise structural roles and extracting specific, qualified outputs from complex hyper-relational facts, even when the underlying information is implicitly present in their training data. The need for exact, structured predictions, rather than descriptive text, appears a key challenge.

F.1 Discussion: LLM Challenges vs. Task-Specific Models for HKGC

Despite their remarkable capabilities across many NLP tasks, directly applying general-purpose LLMs to hyper-relational knowledge graph completion (HKGC) faces significant, distinct challenges. Unlike traditional KGs, HKGs feature complex, high-order structures with qualifiers, leading to unique structural complexities and often higher sparsity. Successfully completing facts by predicting missing elements within these existing complex structures requires precise, structured outputs, a task not inherently aligned with LLMs' text-based, general-purpose nature. While LLMs show promise in related areas like HKG construction from text, their direct efficacy for structured completion within existing HKG facts remains limited.

In contrast, approaches specifically designed for hyper-relational knowledge graph tasks, which we term Task-Specific HKG Models, are built precisely for these challenges. Our proposed HDiff is an example of such a Task-Specific HKG Model. These models are engineered to explicitly capture precise structural patterns, dependencies, and roles within graph data, potentially leveraging various techniques including sequence processing where appropriate for handling representations or specific

components. As observed, HDiff, designed for robust HKGC and trained on HKG structures, accurately predicts precise, qualified entities in the examples, handling name ambiguity and interpreting complex qualifiers to produce structured output. This capability stems from its design which learns to explicitly model intra-fact consistency and leverage structural context.

Therefore, for HKGC demanding accurate, efficient completion within complex, structured data, Task-Specific HKG Models currently offer a more feasible and performant solution than direct application of general LLMs. While LLMs leverage broad knowledge, completing missing elements within complex structures benefits more from models trained explicitly for this task's structural demands. We will explore integrating LLMs into HKGC in future work, potentially for tasks like context generation or noise identification, complementing task-specific models for completion.

G Acronyms and Symbols

This appendix provides reference tables for the acronyms (Table 10) and mathematical symbols (Table 11) used throughout the paper to describe the HDiff framework and related concepts.

Acronym	Description
BT	Barlow Twins (Contrastive Learning Method)
LC	Local Confidence condition
HE	global HyperEdge context condition
CGE	Consistency-Enhanced Global Encoder
CGD	Context-Guided Denoiser
DDPM	Denoising Diffusion Probabilistic Model
GAT	Graph Attention Network
GNN	Graph Neural Network
H-Fact	Hyper-relational Fact
HKG	Hyper-relational Knowledge Graph
HKGC	Hyper-relational Knowledge Graph Completion
KG	Knowledge Graph
KGC	Knowledge Graph Completion
KGE	Knowledge Graph Embedding
MLP	Multi-Layer Perceptron
MRR	Mean Reciprocal Rank

Table 10: List of Acronyms

Symbol	Description	Symbol	Description
$\mathcal{H}$	Hyper-relational knowledge graph	$\mathcal{E}$	Set of entities
$\mathcal{F}_{\text{core}}$	Set of core triples (s, r, o)	$\mathcal{N}'_o(s, r)$	Set of corrupted tail entities for (s, r, o)
f	An H-Fact $(s, r, o, \{(k_i, v_i)\}_{i=1}^n)$	s, r, o	Subject, relation, object
k_i, v_i	i-th qualifier key and value	n	Number of qualifier pairs in an H-Fact
$\mathbf{z}_k$	Intermediate GAT embedding for component k	$\mathbf{z}_f$	GAT embedding for a full H-Fact f
$\mathbf{z}_{sro}$	GAT embedding for core triple (s, r, o)	$\mathbf{p}_f, \mathbf{p}_{sro}$	Projected representations for Barlow Twins
$\mathbf{e}_k$	Final CGE embedding for component k	$\mathbf{e}_f$	Final CGE embedding for an H-Fact f
$\mathbf{E}_E, \mathbf{E}_R, \mathbf{E}_H$	Sets of CGE embeddings for entities, relations, H-Facts	d	Dimensionality of CGE embeddings $\mathbf{e}_k$
$\mathbf{x}_0$	Initial sequence of CGE embeddings for an H-Fact	L	Length of H-Fact sequence $\mathbf{x}_0$
$\mathbf{x}_t$	Noisy H-Fact embedding sequence at timestep t	$\hat{\mathbf{x}}_0$	Estimated clean H-Fact sequence
$\hat{\mathbf{e}}_i$	Estimated clean CGE embedding for component i	$\mathbf{e}_t$	Timestep embedding for diffusion
g_ψ	MLP adaptation layer in CGE	$\mathcal{P}_{\text{BT}}$	Projector network for Barlow Twins
$\mathcal{P}_{\text{conf}}$	MLP for predicting component confidence	Proj_{he}	Projection layer for global context
$\boldsymbol{\epsilon}_\theta$	Conditional denoiser network	$\mathbf{W}_{pred}, \mathbf{b}_{pred}$	Prediction Head weights and bias
$\|\cdot\|_2$	L2 norm (Euclidean norm)	$\mathbf{E}[\cdot]$	Expectation operator
MaxPooling$(\cdot)$	Max pooling operation	sigmoid$(\cdot)$	Sigmoid function
Softmax$(\cdot)$	Softmax function	$\mathbf{I}$	Identity matrix
$\mathcal{L}_{BT}$	Barlow Twins contrastive loss	β_{BT}	Balancing hyperparameter for $\mathcal{L}_{BT}$
$\mathcal{C}$	Cross-correlation matrix for $\mathcal{L}_{BT}$	$\mathcal{L}_{TransE}$	TransE margin-based ranking loss
γ	Margin parameter in TransE loss	$\mathcal{L}_{CGE}$	Total loss for CGE module
$\lambda_{TransE}, \lambda_{BT}$	Loss weights for CGE components	$\mathcal{L}_{diffusion}$	DDPM diffusion loss
$\mathcal{L}_{pred}$	Link prediction loss	λ_{pred}	Weight for prediction loss $\mathcal{L}_{pred}$
$\mathcal{L}$	Overall composite training loss	T	Total number of diffusion timesteps
t	Current diffusion timestep (variable)	$\boldsymbol{\epsilon}$	Gaussian noise vector in diffusion
$\mathbf{c}_{lc}$	Local confidence condition vector	$\mathbf{c}_{he}$	Global H-Fact context condition vector
$p(\mathbf{x}_0)$	Data distribution of clean sequences	$\mathcal{N}(0, \mathbf{I})$	Standard Gaussian distribution
score(i, j)	Attention score (query i, key j)	$\mathbf{q}_i, \mathbf{k}_j$	Query and key vectors in attention
d_k	Dimension of key vectors in attention	$\mathbf{W}_q, \mathbf{W}_c$	Weight matrices for attention biasing

Table 11: List of Symbols