FROM SINGLE- TO MULTI-RELATION: RELATION INCONSISTENCY MODELING FOR UNSUPERVISED GRAPH ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph anomaly detection (GAD) in this work aims to identify anomalous nodes whose behavioral patterns deviate from those of the majority of normal nodes. However, most existing GAD methods are developed for single-relation graphs and cannot directly capture the diverse relational semantics prevalent in real-world systems. Current multi-relation approaches, in turn, largely overlook *cross-relation behavioral inconsistency* as an informative signal of anomalies. To bridge this gap, we identify two complementary forms of relation inconsistency: *explicit inconsistency*, manifested as structural connectivity shifts across relations, and *implicit inconsistency*, revealed by divergent reconstruction patterns. Based on this insight, we propose RelIn-GAD, an architecture-agnostic, plug-and-play module that extends unsupervised reconstruction-based single-relation detectors to multi-relation settings. RelIn-GAD employs global-to-local relation weighting to adaptively aggregate relation-wise reconstruction evidence, and a dual-inconsistency mechanism that distinguishes anomaly-related disparities from benign relational diversity. The resulting inconsistency signal further adapts the node-wise training objective, while explicit structural inconsistency provides a gated correction to reconstruction-based anomaly scoring. Extensive experiments on four real-world datasets and seven backbones demonstrate the broad effectiveness of RelIn-GAD, yielding average relative improvements of up to 36.29% in AUROC and 108.89% in AUPRC, while achieving strong performance against eighteen state-of-the-art baselines.

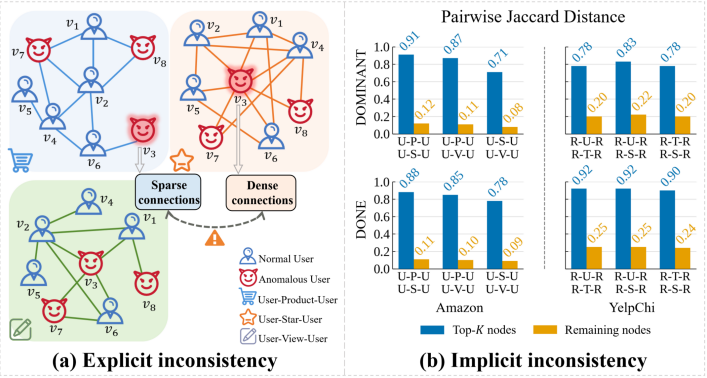
1 INTRODUCTION

Graph anomaly detection (GAD) aims to identify graph instances whose structural connectivity or attribute patterns deviate significantly from those of the majority (Ma et al., 2023; Qiao et al., 2025), with broad applications in financial fraud detection (Cheng et al., 2025; Liu et al., 2022a), cybersecurity intrusion identification (Wu et al., 2025; Hu et al., 2023), and social network analysis (Hooi et al., 2016; Liu et al., 2022b). Existing methods have achieved strong performance in anomalous node detection, yet most target single-relation graphs where nodes interact through a single semantic relation, leaving multi-relation GAD largely underexplored (Guo et al., 2026; Song et al., 2026; Wang et al., 2025b).

In practice, real-world systems such as online review platforms (McAuley & Leskovec, 2013) and financial transaction networks (Xiang et al., 2023) often involve multiple relation types and therefore do not conform to the single-relation assumption (Fu et al., 2025). For instance, a group of users in an online review platform, as shown in Fig. 1(a), can be linked through multiple relations including common purchase behavior (User-Product-User), rating similarity (User-Star-User), and linguistic patterns in reviews (User-View-User), each reflecting a distinct semantic aspect of user behavior. These systems have been widely modeled as multi-relation graphs[1] (Schlichtkrull et al., 2018; Han et al., 2025; Tang et al., 2025). Such graphs exhibit a distinctive property absent in single-relation settings: nodes may behave differently across relations. This motivates us to investigate whether anomalous nodes exhibit characteristic cross-relation behavioral patterns.

[1]We focus on multi-relation graphs where nodes of a single type interact through multiple, well-identified semantic relations. Multi-relation and heterogeneous graphs are parallel, well-established settings, with the latter allowing multiple node types.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Explicit inconsistency (b) Implicit inconsistency

Figure 1: Illustration of two forms of relation inconsistency in multi-relation graphs: (a) *Explicit inconsistency*: anomalous user v_3 appears less connected under User-Product-User yet forms much more direct connections under User-Star-User; and (b) *Implicit inconsistency*: anomalous nodes (i.e., the Top-K nodes ranked by anomaly score) identified by reconstruction-based detectors (DOMINANT and DONE) on individual relation views exhibit limited overlap across relations on Amazon and YelpChi, as evidenced by large pairwise Jaccard distances in a preliminary study in Sec. 4.2.

Translating this insight into a practical multi-relation GAD framework raises two *challenges*: **(1) How to effectively model anomaly-related cross-relation inconsistency?** As illustrated in Fig. 1, we characterize such inconsistency in two complementary forms: (a) *Explicit inconsistency*, where a node exhibits pronounced structural disparity across relations, e.g., being sparsely connected under one relation but densely connected under another; and (b) *Implicit inconsistency*, where a node maintains non-trivial connectivity across relations yet exhibits divergent relation-specific reconstruction behavior. Crucially, different relations may exhibit relation-specific behavioral patterns shared across nodes, making cross-relation variation not inherently anomalous. The key is therefore to identify node-specific inconsistencies beyond such benign relation-induced variation. **(2) How to systematically extend off-the-shelf single-relation GAD detectors to multi-relation graphs without backbone redesign?** Strong reconstruction-based detectors are available for single-relation graphs, yet existing multi-relation GAD methods are typically built with dedicated architectures for cross-relation representation and reconstruction rather than extending these detectors. Their architectural specialization hinders the direct reuse of off-the-shelf single-relation detectors, motivating an architecture-agnostic mechanism that equips them with multi-relation modeling capabilities tailored to GAD.

Existing multi-relation GAD methods do not jointly address these two challenges. For one thing, supervised methods (Han et al., 2025; Tang et al., 2025; Zhang et al., 2021; Peng et al., 2021) explicitly model relation-level differences but heavily depend on labeled data that are expensive to obtain; for another, unsupervised methods (Chen et al., 2023; Li et al., 2025b; Zhai et al., 2026) alleviate label scarcity but use cross-relation information primarily for representation fusion, reconstruction, or consistency regularization, without explicitly separating structural and reconstruction behavioral inconsistencies. Moreover, they do not explicitly incorporate such cross-relation inconsistency as a critical complementary signal for anomaly scoring, despite its potential to enrich the assessment of overall node abnormality. To bridge this gap, we propose RelIn-GAD (**Rel**ation **In**consistency-aware **G**raph **A**nomaly **D**etection), an architecture-agnostic, plug-and-play module for unsupervised multi-relation GAD. RelIn-GAD consists of three core components. First, a graph reconstruction framework provides a common interface for reconstruction-based detectors. Second, a relation inconsistency modeling module combines a multi-relation attention mechanism that learns global-to-local relation importance weights with a dual-inconsistency mechanism that distinguishes anomaly-related inconsistency from benign semantic diversity. Third, the resulting inconsistency score is used to adapt the training objective, while explicit structural inconsistency provides a gated correction to the final anomaly score produced by the backbone. Our *contributions* are summarized as follows:

- We model **cross-relation behavioral inconsistency** from two complementary views: explicit structural disparity across relations and implicit disagreement in relation-specific reconstruction. These signals complement reconstruction-based GAD rather than acting as standalone anomaly criteria.
- To the best of our knowledge, RelIn-GAD is the **first plug-and-play framework** that systematically extends reconstruction-based single-relation GAD detectors to multi-relation graphs without backbone redesign, integrating global-to-local relation weighting with relation inconsistency modeling.
- Extensive experiments show the **broad effectiveness** of RelIn-GAD in improving seven backbone models, while achieving **strong performance** against eighteen unsupervised baseline detectors.

2 Related Work

Single-relation Graph Anomaly Detection. Existing methods for anomalous node detection primarily learn binary or one-class classifiers (Guo et al., 2026; Wang et al., 2025a; Dong et al., 2025; Ma et al., 2024; Shi et al., 2022; Qiao et al., 2024; Qiao & Pang, 2023; Wang et al., 2021; Li et al., 2019), adopt reconstruction-based frameworks (Ding et al., 2019; Bandyopadhyay et al., 2020; Ma et al., 2025; Luo et al., 2022; Fan et al., 2020; Li et al., 2017), or apply contrastive learning (Song et al., 2026; Wang et al., 2025b; Xu et al., 2025; Zhao et al., 2025; Chen et al., 2024; Duan et al., 2024; 2023; Zheng et al., 2023; Liu et al., 2022c). Recent advances in this area have further evolved toward GAD foundation models with the capability to generalize across multiple applications (Pan et al., 2025). Despite their effectiveness, these methods are predominantly designed for single-relation graphs and assume a uniform interaction type among real-world entities, limiting their applicability to multi-relation graphs. To this end, our proposed plug-and-play module enables reconstruction-based methods to be seamlessly extended to unsupervised GAD on multi-relation graphs.

Multi-relation Graph Anomaly Detection. Existing multi-relation methods predominantly focus on semi-supervised settings, typically leveraging relation imbalance-aware classification (Zhang et al., 2021), community-enhanced representation learning (Han et al., 2025), reinforcement learning-based neighbor selection (Peng et al., 2021), or contrastive learning with inter-relation attention (Tang et al., 2025; Zhao et al., 2026). Despite their effectiveness, these methods rely heavily on labeled anomalies, which are often scarce and expensive to obtain in real-world applications. In contrast, unsupervised multi-relation GAD remains relatively underexplored. Existing approaches mainly learn relation-aware reconstruction frameworks (Chen et al., 2023), apply contrastive learning between original and augmented views (Li et al., 2025b), or combine cross-relation representation learning with attention and contrastive objectives (Zhai et al., 2026). However, existing methods do not explicitly distinguish anomaly-related cross-relation inconsistency from benign semantic variation, nor do they systematically exploit structural and relation-specific behavioral inconsistency as complementary anomaly evidence. In contrast, our work explicitly models these complementary forms of relation inconsistency within an unsupervised reconstruction-based framework. A detailed comparison regarding cross-relation formulation, inconsistency modeling, scoring mechanisms, and plug-and-play integration is provided in Appendix C.

3 Methodology

We first formulate the unsupervised multi-relation GAD problem and then detail the core components and objectives of RelIn-GAD, followed by computational complexity analysis in Appendix J. Table 4 in Appendix A summarizes the key notation.

Problem Formulation. Let $\mathcal{G} = \{\mathcal{G}^{(1)}, \ldots, \mathcal{G}^{(R)}\}$ denote a multi-relation graph with R relation types defined over a shared node set $\mathcal{V} = \{v_1, \ldots, v_N\}$. All relations share the same node feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$, while each relation r is characterized by its own edge set $\mathcal{E}^{(r)}$. Accordingly, each relation-specific graph is represented as $\mathcal{G}^{(r)} = (\mathcal{V}, \mathcal{E}^{(r)}, \mathbf{X})$. As depicted in Fig. 2(a), the structure of relation r is represented by an adjacency matrix $\mathbf{A}^{(r)} \in \{0,1\}^{N \times N}$, where $a_{ij}^{(r)} = 1$ if $(v_i, v_j) \in \mathcal{E}^{(r)}$ and $a_{ij}^{(r)} = 0$ otherwise. We consider unsupervised node-level graph anomaly detection. Given $\mathcal{G}$ without label supervision, the objective is to assign each node $v_i \in \mathcal{V}$ an anomaly score s_i, where larger scores reflect stronger anomalous behavior.

3.1 Graph Reconstruction under Residual Analysis

Shared Backbone for Relation-wise Reconstruction. As depicted in Fig. 2(b), the reconstruction backbone employs a shared graph neural network (GNN) encoder f_{Enc} across all relations. For each relation r, the encoder produces q-dimensional node embeddings $\mathbf{Z}^{(r)} = f_{\text{Enc}}(\mathbf{X}, \mathbf{A}^{(r)}) \in \mathbb{R}^{N \times q}$, which are then fed into a relation-specific decoder $f_{\text{Dec}}^{(r)}$ to reconstruct node attributes and graph structure: $\hat{\mathbf{X}}^{(r)}, \hat{\mathbf{A}}^{(r)} = f_{\text{Dec}}^{(r)}(\mathbf{Z}^{(r)})$. The reconstruction error of node v_i under relation r is:

$$e_i^{(r)} = \alpha \cdot \left\| \mathbf{x}_i - \hat{\mathbf{x}}_i^{(r)} \right\|_2 + (1-\alpha) \cdot \left\| \mathbf{a}_i^{(r)} - \hat{\mathbf{a}}_i^{(r)} \right\|_2, \tag{1}$$

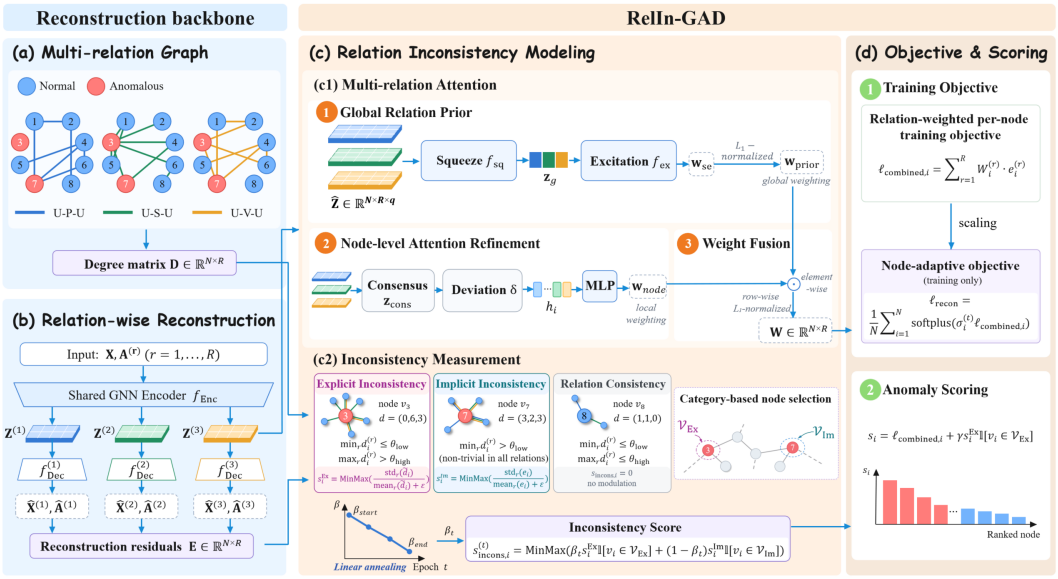


Figure 2: Overview of RelIn-GAD, illustrated on the Amazon dataset as an example. The reconstruction backbone performs relation-wise reconstruction to obtain relation-specific embeddings and residuals (**Left**). RelIn-GAD learns global-to-local relation weights and models explicit structural and implicit reconstruction inconsistencies for node-adaptive training, with explicit inconsistency further providing a gated correction for anomaly scoring (**Right**).

yielding an error matrix $\mathbf{E} \in \mathbb{R}^{N \times R}$. The parameter $\alpha \in [0, 1]$ balances attribute and structural reconstruction. A shared encoder couples training across relation types and places relation-specific representations under a common parameterization, providing a unified basis for comparing and weighting relation-wise reconstruction residuals. For other reconstruction-based GAD backbones, $e_i^{(r)}$ can be directly instantiated by the backbone's native per-node reconstruction residual.

3.2 Relation Inconsistency Modeling

To capture anomaly-related cross-relation deviations beyond benign semantic variation, we introduce a Relation Inconsistency Modeling module , as shown in Fig. 2(c).

3.2.1 Multi-relation Attention

Before measuring cross-relation inconsistency, it is important to account for the fact that different relations may provide unequally informative reconstruction signals for different nodes. Treating all relations equally can cause less reliable relations to dominate or dilute informative deviations, as further analyzed in Appendix D. We therefore introduce a Multi-relation Attention mechanism that learns per-node relation weights $\mathbf{W} \in \mathbb{R}^{N \times R}$ to learn node-specific weights for relation-wise reconstruction residuals. This component proceeds in three steps:

1) Global Relation Prior. Inspired by the squeeze-and-excitation design of SENet (Hu et al., 2018), we design an SE-relation mechanism that captures common cross-relation patterns by deriving a global relation prior $\mathbf{w}_{\text{prior}}$. Specifically, we stack relation-specific embeddings into $\hat{\mathbf{Z}} \in \mathbb{R}^{N \times R \times q}$, with each relation-specific slice independently normalized by $\text{LayerNorm}(\cdot)$ to reduce scale discrepancies across relations. We then perform a relation-wise squeeze operation via global average pooling over the nodes and embedding dimensions: $\mathbf{z}_g = f_{\text{sq}}(\hat{\mathbf{Z}}) \in \mathbb{R}^R$, $z_g^{(r)} = \frac{1}{Nq} \sum_{i=1}^{N} \sum_{k=1}^{q} \hat{Z}_{i,k}^{(r)}$, where $f_{\text{sq}}(\cdot)$ denotes the squeeze function adapted from SENet. Next, an excitation function $f_{\text{ex}}(\cdot)$, following the SENet excitation design and implemented as a two-layer MLP, maps the global relation descriptor $\mathbf{z}_g$ to relation-wise importance scores: $\mathbf{w}_{\text{se}} = f_{\text{ex}}(\mathbf{z}_g)$, The global prior $\mathbf{w}_{\text{prior}}$ is obtained by L_1-normalizing $\mathbf{w}_{\text{se}}$. Since $\mathbf{z}_g$ is aggregated over all nodes, $\mathbf{w}_{\text{prior}}$ captures dataset-level relation importance, while fine-grained node-specific variation is considered in subsequent steps.

2) Node-level Attention Refinement. In anomaly detection, the dominance of normal nodes may bias this global prior toward normal relation patterns. We therefore introduce a node-level MLP to adaptively refine the prior for each node. Specifically, we first compute a prior-weighted consensus: $\mathbf{z}_{\text{cons},i} = \sum_{r=1}^{R} \text{sg}(w_{\text{prior}}^{(r)})\hat{\mathbf{z}}_i^{(r)}$, where $\text{sg}(\cdot)$ denotes the stop-gradient operator. For node v_i, we measure each relation's deviation from the prior-weighted consensus as $\delta_i^{(r)} = \|\hat{\mathbf{z}}_i^{(r)} - \mathbf{z}_{\text{cons},i}\|_2$. The deviation vector $\boldsymbol{\delta}_i$ explicitly captures relation-wise disagreement with the consensus. We concatenate it with the flattened relation embeddings: $\mathbf{h}_i = \left[\text{vec}(\hat{\mathbf{Z}}_{i,:,:}) \parallel \boldsymbol{\delta}_i\right] \in \mathbb{R}^{R(q+1)}$. A two-layer MLP with dropout and temperature-scaled softmax produces: $\mathbf{w}_{\text{node},i} = \text{softmax}\left(\frac{\mathbf{U}_2\,\text{ReLU}(\mathbf{U}_1\mathbf{h}_i)}{\tau_t}\right)$, where $\tau_t = \max(\tau_{\min}, \tau_0\lambda^t)$ is an annealed temperature that encourages smoother weights early in training and sharper relation selection later.

3) Weight Fusion. Finally, we combine the dataset-level relation prior with the node-level refinement. Specifically, for each node v_i, we combine $\mathbf{w}_{\text{prior}}$ and $\mathbf{w}_{\text{node},i}$ through element-wise multiplication: $\tilde{\mathbf{W}}_{i,:} = (\mathbf{w}_{\text{prior}} \odot \mathbf{w}_{\text{node},i})^\top$. We then apply row-wise L_1 normalization to obtain the final node-specific relation weight matrix $\mathbf{W} \in \mathbb{R}^{N\times R}$. Appendix H further analyzes how the learned relation weights adapt to varying relation quality.

3.2.2 Inconsistency Measurement

While the preceding attention mechanism determines the relative contribution of relation-wise reconstruction residuals, but does not assess whether cross-relation disagreement provides reliable anomaly-related evidence. As observed in Fig. 1(b), the top-K high-scoring nodes identified from individual relation views exhibit limited overlap, indicating that relation-wise reconstruction responses can differ substantially across relations; however, such variation may also arise from benign relation-specific semantics. We therefore introduce an Inconsistency Measurement to identify reliable node-specific cross-relation deviations.

Before quantifying cross-relation inconsistency, we first determine which type of evidence is reliable for each node. Structural inconsistency can be directly assessed when relation-wise connectivity exhibits pronounced contrast, whereas reconstruction behavioral inconsistency is meaningful only when the node is sufficiently represented across all relation views. When neither condition provides reliable cross-relation evidence, we avoid applying inconsistency-dependent modulation.

Accordingly, we partition nodes into three mutually exclusive categories based on their cross-relation connectivity patterns. Let $\mathbf{D} = [d_i^{(r)}] \in \mathbb{R}^{N\times R}$ denote the relation-wise degree matrix, where $d_i^{(r)}$ is the degree of node v_i under relation r, and $\theta_{\text{low}} < \theta_{\text{high}}$ denote the degree thresholds. By default, we set $\theta_{\text{low}} = 0$ and $\theta_{\text{high}} = \frac{1}{N}\sum_{i=1}^{N} \max_r d_i^{(r)}$.

Definition 1 (Explicit Inconsistency) *Node v_i is **explicit-inconsistent** if $\min_r d_i^{(r)} \le \theta_{\text{low}}$ and $\max_r d_i^{(r)} > \theta_{\text{high}}$. Such a node is sparsely connected under at least one relation while densely connected under another.*

Definition 2 (Implicit Inconsistency) *Node v_i is **implicit-inconsistent** if $\min_r d_i^{(r)} > \theta_{\text{low}}$. Such a node maintains non-trivial connectivity across all relations, making cross-relation reconstruction comparison meaningful.*

Definition 3 (Relation Consistency) *Node v_i is **relation-consistent** if $\min_r d_i^{(r)} \le \theta_{\text{low}}$ and $\max_r d_i^{(r)} \le \theta_{\text{high}}$. Such a node exhibits low connectivity under at least one relation, while showing no sufficiently high connectivity under any relation.*

Applying these criteria to the degree matrix $\mathbf{D}$ yields the explicit- and implicit-inconsistent node sets $\mathcal{V}_{\text{Ex}}$ and $\mathcal{V}_{\text{Im}}$, respectively, while relation-consistent nodes are excluded from inconsistency-dependent modulation. Based on this taxonomy, we define category-specific inconsistency scores. Structural inconsistency is quantified from $\mathbf{D}$, whereas implicit inconsistency is evaluated from the reconstruction residual matrix $\mathbf{E}$. For explicit-inconsistent nodes, we first normalize relation-wise degrees to account for differences in relation density. Let $\bar{d}^{(r)} = \frac{1}{N}\sum_{j=1}^{N} d_j^{(r)}$ and $\tilde{d}_i^{(r)} =$

$d_i^{(r)}/(\bar{d}^{(r)}+1)$. We then quantify structural disparity across relations as:

$$s_i^{\mathrm{Ex}} = \mathrm{MinMax}\left(\frac{\mathrm{std}_r(\tilde{\mathbf{d}}_i)}{\mathrm{mean}_r(\tilde{\mathbf{d}}_i)+\varepsilon}\right), \tag{2}$$

where $\tilde{\mathbf{d}}_i = [\tilde{d}_i^{(1)}, \ldots, \tilde{d}_i^{(R)}]$ and $\mathrm{MinMax}(\cdot)$ denotes min–max normalization over the node set.

Analogically, for implicit-inconsistent nodes, we quantify reconstruction disagreement as:

$$s_i^{\mathrm{Im}} = \mathrm{MinMax}\left(\frac{\mathrm{std}_r(\mathbf{e}_i)}{\mathrm{mean}_r(\mathbf{e}_i)+\varepsilon}\right), \tag{3}$$

where $\mathbf{e}_i = [e_i^{(1)}, \ldots, e_i^{(R)}]$ collects the relation-specific residuals from Eq. (1). Both scores are therefore mapped into the range $[0, 1]$ and are directly comparable across the two categories.

Since reconstruction errors are less reliable early in training, we linearly anneal $\beta_t = \beta_{\mathrm{end}} + (\beta_{\mathrm{start}} - \beta_{\mathrm{end}})\big(1 - \frac{t}{T}\big)$, where T is the total number of training epochs and $0 \le \beta_{\mathrm{end}} \le \beta_{\mathrm{start}} \le 1$. The final inconsistency score is

$$s_{\mathrm{incons},i}^{(t)} = \mathrm{MinMax}\big(\beta_t s_i^{\mathrm{Ex}} \mathbb{I}[v_i \in \mathcal{V}_{\mathrm{Ex}}] + (1-\beta_t) s_i^{\mathrm{Im}} \mathbb{I}[v_i \in \mathcal{V}_{\mathrm{Im}}]\big), \tag{4}$$

where $\mathbb{I}[\cdot]$ denotes the indicator function, and $\mathcal{V}_{\mathrm{Ex}}$ and $\mathcal{V}_{\mathrm{Im}}$ denote the sets of explicit- and implicit-inconsistent nodes, respectively. A larger β_t emphasizes explicit structural inconsistency early in training, when reconstruction errors are less reliable; as β_t decreases, the emphasis gradually shifts toward implicit reconstruction inconsistency.

3.3 Training Objective and Anomaly Scoring

1) Training Objective. We first aggregate the relation-wise residuals into a relation-weighted per-node objective:

$$\ell_{\mathrm{combined},i} = \sum\nolimits_{r=1}^{R} W_i^{(r)} \cdot e_i^{(r)}. \tag{5}$$

While relation weighting controls the contribution of different relation views, the inconsistency score $s_{\mathrm{incons},i}^{(t)}$ characterizes the strength of reliable cross-relation disagreement for node v_i. We therefore use the latter to modulate each node's contribution during training. Specifically, we define $\sigma_i^{(t)} = 1 + \mathrm{sg}\Big(s_{\mathrm{incons},i}^{(t)}\Big)(\sigma_{\max} - 1)$, where $\sigma_{\max}$ controls the strength of inconsistency-aware scaling. The stop-gradient operator $\mathrm{sg}(\cdot)$ treats the inconsistency score as a fixed node-wise coefficient during backpropagation, and the resulting node-adaptive training objective is given as:

$$\ell_{\mathrm{recon}} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{softplus}\Big(\sigma_i^{(t)} \ell_{\mathrm{combined},i}\Big). \tag{6}$$

Consequently, the scaling adjusts each node's gradient magnitude without propagating gradients through the inconsistency estimation procedure. This modulation is applied only during training, placing greater optimization emphasis on nodes with higher inconsistency scores. Its effect is further analyzed in Appendix E.1.

2) Anomaly Scoring. At inference, the anomaly score is computed from the attention-weighted reconstruction error. For explicit-inconsistent nodes, we further introduce a gated structural correction:

$$s_i = \ell_{\mathrm{combined},i} + \gamma s_i^{\mathrm{Ex}} \mathbb{I}[v_i \in \mathcal{V}_{\mathrm{Ex}}], \tag{7}$$

where γ controls the correction strength. The additional term supplements reconstruction evidence when pronounced cross-relation structural disparity is present. Further discussion of this gated correction is provided in Appendix E.2.

4 Experiments

Our experiments include a preliminary study on anomaly-related implicit relation inconsistency, baseline comparisons, key component ablations, additional analyses of case studies (Appendix F), and sensitivity analyses (Appendix G). We further examine the functionality of adaptive weighting in Appendix D and study relation weight behavior under controlled relation perturbations in Appendix H.

4.1 EXPERIMENTAL SETUP

Datasets. In our experiments, we employ four real-world multi-relation GAD datasets spanning online review fraud and financial transaction fraud: Amazon (McAuley & Leskovec, 2013), YelpChi (Rayana & Akoglu, 2015), S-FFSD (Xiang et al., 2023), and IEEE-CIS (Ge et al., 2020). This experimental protocol provides diverse evaluation settings with different node attributes, relation semantics, graph scales, and anomaly ratios. Appendix I further introduces these datasets, details data split, and provides dataset statistics in Table 11.

Baselines. We compare RelIn-GAD with eighteen state-of-the-art unsupervised GAD methods spanning diverse methodological categories: *Reconstruction-based*: DOMINANT (Ding et al., 2019), AnomalyDAE (Fan et al., 2020), DONE (Bandyopadhyay et al., 2020), GAD-NR (Roy et al., 2024), ADA-GAD (He et al., 2024), DiffGAD (Li et al., 2025a), and CoCo (Wang et al., 2025a); *Contrastive Learning-based*: CoLA (Liu et al., 2022c), GRADATE (Duan et al., 2023), ARISE (Duan et al., 2024), GADAM (Chen et al., 2024), AD-GCL (Xu et al., 2025), and GADACE (Wang et al., 2025b); *Affinity-based*: TAM (Qiao & Pang, 2023) and FreeGAD (Zhao et al., 2025); *Spectral GNN-based*: SmoothGNN (Dong et al., 2025); and *Multi-relational*: UMGAD (Li et al., 2025b) and SFGA (Zhai et al., 2026). Our experiments use their official open-sourced implementations. We further treat the seven reconstruction-based methods as backbone models and evaluate each in two settings: its vanilla form and an augmented variant equipped with RelIn-GAD, allowing a direct assessment of RelIn-GAD's plug-and-play applicability and generalizability.

Implementation Details. For a fair comparison, each vanilla backbone and its RelIn-GAD-augmented counterpart share the same backbone-specific configurations. For parameters of RelIn-GAD, we use grid search as detailed in Appendix G. For the remaining baselines, we adopt the default hyperparameters reported in their respective publications and open-source implementations. All experiments are conducted on a server equipped with an NVIDIA GeForce RTX 3090 GPU (24GB RAM). The code for RelInGAD is available at `https://anonymous.4open.science/r/RelIn-GAD-D7F5`. Following a widely used benchmark (Tang et al., 2023), we report the mean and standard deviation of Area Under the ROC Curve (AUROC) and Area Under the Precision-Recall Curve (AUPRC) over five independent runs.

4.2 PRELIMINARY STUDY: DIVERGENT RECONSTRUCTION PATTERNS ACROSS RELATIONS

To verify the existence of implicit inconsistency in multi-relation graphs, we conduct a preliminary study with two representative unsupervised reconstruction-based detectors, DOMINANT (Ding et al., 2019) and DONE (Bandyopadhyay et al., 2020) on Amazon (McAuley & Leskovec, 2013) and YelpChi (Rayana & Akoglu, 2015). For each relation r, we independently apply each detector, rank nodes by their anomaly scores, and select the top-K nodes as the relation-specific anomaly set $\mathcal{A}^{(r)}$, with the remaining $N-K$ nodes forming the normal set $\mathcal{N}^{(r)}$, where K is the number of ground-truth anomalous nodes (K=821 for Amazon and K=6,677 for YelpChi). As shown in Fig. 1(b), the Jaccard distances between relation-specific anomaly sets are consistently high across all relation pairs and both detectors, whereas those between normal sets remain low. This pronounced asymmetry indicates that different relations identify substantially different subsets of anomalous nodes while largely agreeing on normal nodes. These observations provide empirical evidence for implicit inconsistency across relations, motivating its explicit modeling in our method. The definition of Jaccard distance is provided in Appendix B.

4.3 EFFECTIVENESS ON RECONSTRUCTION-BASED GAD BACKBONES

This study evaluates the effectiveness of RelIn-GAD when integrated with reconstruction-based GAD backbones on multi-relation graphs. For each baseline, we consider two settings: 1) a *vanilla* version that follows existing works (Li et al., 2025b; Zhai et al., 2026) by treating the multi-relation graph as a single-view graph; and 2) an *augmented* version (+RelIn-GAD) that integrates our RelIn-GAD module to incorporate relation-specific information while keeping the backbone architecture unchanged. For each backbone, the vanilla and augmented variants use identical training configurations, including the learning rate, embedding dimension, and number of training epochs. As shown in Table 1, integrating RelIn-GAD improves the performance of reconstruction-based backbones in most cases, with the largest gains observed on Amazon, where relation inconsistency is particularly pronounced. On YelpChi, S-FFSD and IEEE-CIS, where the various relationships are more consistent, the absolute

Table 1: Comparison of reconstruction-based models with or without incorporating the RelIn-GAD module w.r.t. AUROC and AUPRC (%).

Method	Version	Amazon		YelpChi		S-FFSD		IEEE-CIS		Avg. Improvement	
		AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
DOMINANT	Vanilla	64.07±0.22	9.43±0.27	54.31±1.05	16.96±1.19	66.68±0.25	31.73±0.40	64.05±2.67	13.44±1.69	+11.90%	+65.61%
	+RelIn-GAD	86.30±0.35	32.47±0.73	58.75±1.53	20.04±2.79	68.33±0.44	32.82±0.49	65.49±1.67	12.97±0.74		
AnomalyDAE	Vanilla	83.40±1.66	23.93±2.17	61.75±0.14	23.19±0.13	63.11±0.86	26.99±2.02	67.74±0.45	16.00±0.97	+2.76%	+16.00%
	+RelIn-GAD	88.53±0.34	35.58±3.08	63.68±0.08	25.23±0.09	63.97±0.39	27.96±1.33	68.00±0.48	16.47±0.16		
DONE	Vanilla	64.01±4.41	8.78±1.20	48.81±0.57	14.35±0.38	36.61±0.62	13.80±0.10	57.96±1.74	10.19±0.32	+36.29%	+108.89%
	+RelIn-GAD	84.17±1.77	24.31±3.35	61.08±1.29	20.56±0.41	64.96±1.14	38.11±1.38	64.39±2.68	14.19±1.11		
GAD-NR	Vanilla	52.40±2.51	6.94±0.31	51.55±5.78	16.75±2.65	44.08±5.99	25.89±1.01	61.41±1.17	13.16±0.80	+10.41%	+1.39%
	+RelIn-GAD	58.32±4.43	9.31±2.04	59.07±1.53	18.96±0.76	49.78±8.47	19.22±7.45	63.14±1.55	11.05±0.53		
ADA-GAD	Vanilla	23.04±0.11	4.10±0.00	51.17±0.45	15.21±0.21	60.21±0.65	21.57±0.72	63.25±1.09	14.24±0.83	+15.73%	+11.91%
	+RelIn-GAD	32.62±4.50	4.88±0.68	61.88±2.11	19.48±1.25	57.58±1.78	20.23±0.91	66.27±1.17	15.20±1.17		
DiffGAD	Vanilla	68.42±0.15	10.40±0.08	58.20±0.11	22.59±0.07	59.46±1.05	23.81±0.81	61.97±0.30	13.03±0.10	+8.16%	+38.99%
	+RelIn-GAD	83.73±1.01	24.83±2.28	61.05±1.12	24.04±0.44	62.66±1.92	26.07±2.13	61.95±1.20	13.20±0.48		
CoCo	Vanilla	84.05±1.67	23.00±4.01	55.06±1.20	17.06±0.51	33.23±1.17	13.21±0.22	48.10±0.16	8.71±0.10	+19.47%	+41.02%
	+RelIn-GAD	80.82±3.05	20.58±4.56	62.26±0.92	21.04±0.26	56.03±0.90	32.92±0.33	48.11±0.07	8.89±0.16		

Table 2: Comparison between RelIn-GAD and 18 baselines w.r.t. AUROC and AUPRC (%). The best results are highlighted in red, and the second best are in blue.

	Method	Amazon		YelpChi		S-FFSD		IEEE-CIS	
		AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
Recon.	DOMINANT [SDM'19]	64.07±0.22	9.43±0.27	54.31±1.05	16.96±1.19	66.68±0.25[1]	31.73±0.40[1]	64.05±2.67	13.44±1.69
	AnomalyDAE [ICASSP'20]	83.40±1.66	23.93±2.17	61.75±0.14	23.19±0.13	63.11±0.86	26.99±2.02	67.74±0.45	16.00±0.97
	DONE [WSDM'20]	64.01±4.41	8.78±1.20	48.81±0.57	14.35±0.38	36.61±0.62	13.80±0.10	57.96±1.74	10.19±0.32
	GAD-NR [WSDM'24]	52.40±2.51	6.94±0.31	51.55±5.78	16.75±2.65	44.08±5.99	25.89±1.01	61.41±1.17	13.16±0.80
	ADA-GAD [AAAI'24]	23.04±0.11	4.10±0.00	51.17±0.45	15.21±0.21	60.21±0.65	21.57±0.72	63.25±1.09	14.24±0.83
	DiffGAD [ICLR'25]	68.42±0.15	10.40±0.08	58.20±0.11	22.59±0.07	59.46±1.05	23.81±0.81	61.97±0.30	13.03±0.10
	CoCo [TKDE'25]	84.05±1.67	23.00±4.01	55.06±1.20	17.06±0.51	33.23±1.17	13.21±0.22	48.10±0.16	8.71±0.10
Cont.	CoLA [TNNLS'21]	17.18±0.43	4.26±0.05	47.58±0.20	13.35±0.08	36.51±0.41	14.63±0.15	44.63±0.72	7.46±0.15
	GRADATE [AAAI'23]	35.39±1.36	5.29±0.19	50.82±1.25	14.87±0.60	36.07±1.23	14.45±0.24	47.23±0.91	7.76±0.13
	ARISE [TNNLS'24]	40.72±0.00	6.63±0.00	49.30±0.54	14.30±0.27	49.57±0.00	17.72±0.00	50.32±0.42	8.59±0.06
	GADAM [ICLR'24]	44.58±2.05	5.69±0.29	47.69±0.88	13.42±0.29	36.59±3.90	14.05±0.60	47.68±2.74	8.07±0.63
	AD-GCL [AAAI'25]	18.63±1.93	4.16±0.17	45.39±0.81	13.07±0.27	47.97±4.22	17.15±1.38	51.36±2.77	9.25±0.83
	GADACE [ICASSP'25]	53.27±1.68	7.00±0.36	44.99±1.06	12.78±0.36	40.39±0.72	14.39±0.15	53.50±0.14	9.63±0.09
Affi.	TAM [NeurIPS'23]	74.50±1.39	19.16±1.70	45.92±0.37	13.59±0.20	36.43±0.26	14.29±0.12	46.68±0.31	7.85±0.02
	FreeGAD [CIKM'25]	86.73±0.00	69.33±0.00	45.60±0.00	12.80±0.00	50.81±0.00	16.67±0.00	43.33±0.00	7.42±0.00
Spec.	SmoothGNN [WWW'25]	53.25±15.86	8.20±2.15	49.99±0.59	14.53±0.15	45.62±20.49	21.46±11.42	51.49±2.07	8.88±0.44
Multi.	UMGAD [ICDE'25]	87.84±0.28	32.36±2.29	59.92±0.23	22.98±0.21	62.19±2.73	26.03±4.58	66.25±0.84	14.30±0.49
	SFGA [AAAI'26]	71.45±9.30	13.25±3.10	48.16±0.28	13.75±0.12	55.03±0.16	21.66±0.50	51.01±0.73	9.01±0.12
	RelIn-GAD + AnomalyDAE	88.53±0.34	35.58±3.08	63.68±0.08	25.23±0.09	63.97±0.39	27.96±1.33	68.00±0.48	16.47±0.16

[1] RelIn-GAD with DOMINANT backbone achieves higher values in both AUROC and AUPRC, as shown in Table 1.

improvements are smaller. This observation is consistent with the role of relation inconsistency as complementary anomaly evidence. When cross-relation inconsistency is less pronounced, explicitly modeling it naturally provides a smaller additional benefit. Overall, these results demonstrate the plug-and-play applicability of RelIn-GAD across diverse reconstruction-based backbones, extending them to multi-relation graphs without requiring backbone redesign.

4.4 Comparison with Existing Unsupervised GAD Baselines

To evaluate whether RelIn-GAD can enable a single-relation backbone to achieve competitive performance in multi-relation settings, we compare a representative configuration (RelIn-augmented AnomalyDAE) against diverse unsupervised baselines described in Sec. 4.1. As reported in Table 2, RelIn-GAD achieves the superior overall performance over other baselines. Compared with the state-of-the-art multi-relation GAD methods, our approach achieves significant performance gains, with the gap widening on datasets with stronger relation inconsistency. More importantly, it empowers established single-relation detection techniques to thrive in complex multi-relation scenarios. To ensure a fair comparison, we use the same values for hyperparameters that serve equivalent roles across multi-relation models. For example, the balance parameter α in Eq. (1), which controls the trade-off between attribute and structural reconstruction, is set to 0.6 on Amazon and 0.5 on the others.

Table 3: Comparison between RelIn-GAD and five ablated variants w.r.t. AUROC and AUPRC (%)

Method	Amazon		YelpChi		S-FFSD		IEEE-CIS	
	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC	AUPRC
w/o SE	86.38±3.49	31.16±5.62	63.57±0.09	25.04±0.13	63.27±2.62	27.91±3.43	67.00±0.70	15.91±1.45
w/o NA	85.75±1.44	25.78±2.83	**63.95±0.17**	**25.51±0.23**	63.58±0.94	28.25±1.88	67.33±0.85	16.07±1.16
w/o DP	82.88±6.89	25.62±5.30	63.64±0.15	25.22±0.21	63.29±0.71	26.56±1.58	66.74±1.29	15.30±1.14
w/o EX	85.79±1.44	30.50±4.43	63.63±0.18	25.15±0.32	**63.99±1.07**	**28.86±2.31**	68.00±0.48	16.47±0.16
w/o IM	83.83±2.62	30.01±3.66	63.61±0.15	25.22±0.08	63.05±0.64	26.51±1.34	68.00±0.48	16.47±0.16
RelIn-GAD	**88.53±0.34**	**35.58±3.08**	63.68±0.08	25.23±0.09	63.97±0.39	27.96±1.33	**68.00±0.48**	**16.47±0.16**

4.5 Ablation Studies

To assess the contribution of each design component, we conduct ablation studies with five variants of RelIn-GAD, each removing one component while keeping the others unchanged: *w/o SE* removes the SE attention block and adopts uniform weights of $1/R$; *w/o NA* removes the node-level attention—the two-step iterative refinement, and only uses the SE prior as the final attention weights; *w/o DP* removes the deviation from the prior-weighted consensus, so that the node-level attention takes only $\mathrm{vec}(\hat{\mathbf{Z}}_{i,:,:})$ as input instead of $\mathbf{h}_i$; and *w/o EX* and *w/o IM* exclude the explicit and implicit inconsistency signals, respectively, by setting $s_i^{\mathrm{Ex}} \equiv 0$ or $s_i^{\mathrm{Im}} \equiv 0$.

As observed in Table 3, overall, removing any individual component degrades performance on at least one dataset, supporting the role of the proposed components across diverse multi-relation graphs. On Amazon, removing any component consistently degrades performance, confirming that both explicit and implicit inconsistency cues are beneficial. On YelpChi, *w/o NA* slightly improves AUROC by 0.27 percentage points. This is because its relations capture highly correlated review behaviors, limiting the benefit of node-specific weighting. On S-FFSD, *w/o EX* slightly improves AUROC by 0.02 percentage points. As shown by the dataset statistics in Appendix I, anomalous nodes exhibit consistently sparse connectivity across relations, with average degrees below 4. Consequently, explicit connectivity-based cross-relation inconsistency provides limited discriminative information on this dataset. These observations are consistent with our treatment of relation inconsistency as complementary rather than universally informative evidence. RelIn-GAD does not assume that every inconsistency cue contributes equally across datasets; when a particular cross-relation signal is weak, removing its corresponding component may have little effect or occasionally yield a small improvement. Nevertheless, the full framework provides consistent overall improvements by jointly combining relation-aware reconstruction with complementary cross-relation cues.

4.6 Relation Weight Adaptation under Relation Quality Variation

To examine whether the learned relation weights reflect relation quality rather than fixed relation identity, we conduct controlled structural perturbations on Amazon, with detailed settings and results provided in Appendix H. When a relation is made highly correlated or redundant with another relation, its learned weight tends to shift toward that relation's weight; in contrast, introducing noisy or missing edges more often decreases its weight. These results suggest that the weighting mechanism adapts to relation-specific structural similarity and degradation, rather than assigning fixed importance to relation types.

5 Conclusion

In this work, we identify cross-relation behavioral inconsistency as an informative signal for anomaly detection on multi-relation graphs and propose RelIn-GAD to effectively capture and exploit this signal. As a plug-and-play module, RelIn-GAD extends reconstruction-based single-relation detectors to multi-relation settings without requiring backbone redesign. Extensive experiments on real-world datasets demonstrate that RelIn-GAD consistently improves diverse reconstruction-based detectors, with particularly strong gains on datasets exhibiting pronounced cross-relation inconsistency. Building on these findings, a natural direction for future work is to investigate relation inconsistency in temporal multi-relation graphs, where relation-specific interactions evolve over time, while retaining the plug-and-play design of RelIn-GAD.

AI USE STATEMENT

In this work, we have not used generative AI tools for tasks with required disclosure. Additionally, we used generative AI tools for literature search and for editing the manuscript to improve readability and linguistic clarity, which are tasks with recommended disclosure. We have reviewed all AI-assisted work, and all identified references were independently verified against their original sources before inclusion. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work proposes RelIn-GAD, an architecture-agnostic, plug-and-play module that extends unsupervised reconstruction-based single-relation graph anomaly detectors to multi-relation settings. Our experiments are conducted exclusively on publicly available benchmark datasets, including Amazon, YelpChi, S-FFSD, and IEEE-CIS. This work does not involve human-subject studies, collection of new user data, or the use of sensitive demographic attributes. No ethical approval or IRB review was required for this study. The authors declare no conflicts of interest or external sponsorship that could have influenced the research.

REPRODUCIBILITY STATEMENT

We have taken several measures to ensure the reproducibility of our work. The main paper and appendix provide the model formulation, training objectives, dataset and preprocessing descriptions, implementation details, hyperparameter settings, experimental protocols, and additional analyses necessary to reproduce the reported results. We additionally provide anonymized source code as supplementary material, including the implementation and configurations used in our experiments. Together, the paper, appendix, and supplementary code provide the information necessary to reproduce the proposed method and experimental results.

REFERENCES

Sambaran Bandyopadhyay, Lokesh N, Saley Vishal Vivek, and M. N. Murty. Outlier resistant unsupervised deep architectures for attributed network embedding. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pp. 25–33, 2020.

Jingyan Chen, Guanghui Zhu, Chunfeng Yuan, and Yihua Huang. Boosting graph anomaly detection with adaptive message passing. In *The Twelfth International Conference on Learning Representations*, 2024.

Ling-Hao Chen, He Li, Wanyuan Zhang, Jianbin Huang, Xiaoke Ma, Jiangtao Cui, Ning Li, and Jaesoo Yoo. AnomMAN: Detect anomalies on multi-view attributed networks. *Information Sciences*, 628:1–21, 2023.

Dawei Cheng, Yao Zou, Sheng Xiang, and Changjun Jiang. Graph neural networks for financial fraud detection: A review. *Frontiers of Computer Science*, 19(9):199609, 2025.

Kaize Ding, Jundong Li, Rohit Bhanushali, and Huan Liu. Deep anomaly detection on attributed networks. In *Proceedings of the 2019 SIAM International Conference on Data Mining (SDM)*, pp. 594–602, 2019.

Xiangyu Dong, Xingyi Zhang, Yanni Sun, Lei Chen, Mingxuan Yuan, and Sibo Wang. SmoothGNN: Smoothing-aware gnn for unsupervised node anomaly detection. In *Proceedings of the ACM on Web Conference 2025*, pp. 1225–1236, 2025.

Jingcan Duan, Siwei Wang, Pei Zhang, En Zhu, Jingtao Hu, Hu Jin, Yue Liu, and Zhibin Dong. Graph anomaly detection via multi-scale contrastive learning networks with augmented view. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(6):7459–7467, 2023.

Jingcan Duan, Bin Xiao, Siwei Wang, Haifang Zhou, and Xinwang Liu. ARISE: Graph anomaly detection on attributed networks via substructure awareness. *IEEE Transactions on Neural Networks and Learning Systems*, 35(12):18172–18185, 2024.

Haoyi Fan, Fengbin Zhang, and Zuoyong Li. Anomalydae: Dual autoencoder for anomaly detection on attributed networks. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5685–5689, 2020.

Chao Fu, Guannan Liu, Kun Yuan, and Junjie Wu. Nowhere to H^2IDE: Fraud detection from multi-relation graphs via disentangled homophily and heterophily identification. *IEEE Transactions on Knowledge and Data Engineering*, 37(3):1380–1393, 2025.

Dingling Ge, Jianyang Gu, Shunyu Chang, and JingHui Cai. Credit card fraud detection using lightgbm model. In *2020 International Conference on E-Commerce and Internet Technology (ECIT)*, pp. 232–236, 2020.

Ting Guo, Dongyu Pei, Gangzhu Qiao, Kaixuan Yao, and Da Wang. PAGE: Progressive anomaly generation network for semi-supervised graph anomaly detection. In *Proceedings of the ACM Web Conference 2026*, pp. 959–967, 2026.

Li Han, Longxun Wang, Ziyang Cheng, Bo Wang, Guang Yang, Dawei Cheng, and Xuemin Lin. Mitigating the tail effect in fraud detection by community enhanced multi-relation graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 37(4):2029–2041, 2025.

Junwei He, Qianqian Xu, Yangbangyan Jiang, Zitai Wang, and Qingming Huang. ADA-GAD: anomaly-denoised autoencoders for graph anomaly detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(8):8481–8489, 2024.

Bryan Hooi, Hyun Ah Song, Alex Beutel, Neil Shah, Kijung Shin, and Christos Faloutsos. FRAUDAR: Bounding graph fraud in the face of camouflage. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 895–904, 2016.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.

Xiaoyan Hu, Wenjie Gao, Guang Cheng, Ruidong Li, Yuyang Zhou, and Hua Wu. Toward early and accurate network intrusion detection using graph embedding. *IEEE Transactions on Information Forensics and Security*, 18:5817–5831, 2023.

Jinghan Li, Yuan Gao, Jinda Lu, Junfeng Fang, Congcong Wen, Hui Lin, and Xiang Wang. Diff-GAD: A diffusion-based unsupervised graph anomaly detector. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Jundong Li, Harsh Dani, Xia Hu, and Huan Liu. Radar: Residual analysis for anomaly detection in attributed networks. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pp. 2152–2158, 2017.

Xiang Li, Jianpeng Qi, Zhongying Zhao, Guanjie Zheng, Lei Cao, Junyu Dong, and Yanwei Yu. UMGAD: Unsupervised multiplex graph anomaly detection. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, pp. 3724–3737, 2025b.

Yuening Li, Xiao Huang, Jundong Li, Mengnan Du, and Na Zou. SpecAE: Spectral autoencoder for anomaly detection in attributed networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 2233–2236, 2019.

Fanzhen Liu, Zhao Li, Baokun Wang, Jia Wu, Jian Yang, Jiaming Huang, Yiqing Zhang, Weiqiang Wang, Shan Xue, Surya Nepal, and Quan Z. Sheng. eRiskCom: An e-commerce risky community detection platform. *The VLDB Journal*, 31:1085–1101, 2022a.

Fanzhen Liu, Xiaoxiao Ma, Jia Wu, Jian Yang, Shan Xue, Amin Beheshti, Chuan Zhou, Hao Peng, Quan Z. Sheng, and Charu C. Aggarwal. DAGAD: Data augmentation for graph anomaly detection. In *2022 IEEE International Conference on Data Mining (ICDM)*, pp. 259–268, 2022b.

Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE Transactions on Neural Networks and Learning Systems*, 33(6):2378–2392, 2022c.

Xuexiong Luo, Jia Wu, Amin Beheshti, Jian Yang, Xiankun Zhang, Yuan Wang, and Shan Xue. ComGA: Community-aware attributed graph anomaly detection. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 657–665, 2022.

Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z. Sheng, Hui Xiong, and Leman Akoglu. A comprehensive survey on graph anomaly detection with deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12012–12038, 2023.

Xiaoxiao Ma, Ruikun Li, Fanzhen Liu, Kaize Ding, Jian Yang, and Jia Wu. Graph anomaly detection with few labels: A data-centric approach. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2153–2164, 2024.

Xiaoxiao Ma, Fanzhen Liu, Jia Wu, Jian Yang, Shan Xue, and Quan Z. Sheng. Rethinking unsupervised graph anomaly detection with deep learning: Residuals and objectives. *IEEE Transactions on Knowledge and Data Engineering*, 37(2):881–895, 2025.

Julian John McAuley and Jure Leskovec. From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd International Conference on World Wide Web*, pp. 897–908, 2013.

Junjun Pan, Yu Zheng, Yue Tan, and Yixin Liu. A survey of generalization of graph anomaly detection: From transfer learning to foundation models. In *2025 IEEE International Conference on Knowledge Graph (ICKG)*, pp. 316–323, 2025.

Hao Peng, Ruitong Zhang, Yingtong Dou, Renyu Yang, Jingyi Zhang, and Philip S. Yu. Reinforced neighborhood selection guided multi-relational graph neural networks. *ACM Transactions on Information Systems*, 40(4):1–46, 2021.

Hezhe Qiao and Guansong Pang. Truncated affinity maximization: One-class homophily modeling for graph anomaly detection. In *Advances in Neural Information Processing Systems*, volume 36, pp. 49490–49512, 2023.

Hezhe Qiao, Qingsong Wen, Xiaoli Li, Ee-Peng Lim, and Guansong Pang. Generative semi-supervised graph anomaly detection. In *Advances in Neural Information Processing Systems*, volume 37, pp. 4660–4688, 2024.

Hezhe Qiao, Hanghang Tong, Bo An, Irwin King, Charu Aggarwal, and Guansong Pang. Deep graph anomaly detection: A survey and new perspectives. *IEEE Transactions on Knowledge and Data Engineering*, 37(9):5106–5126, 2025.

Shebuti Rayana and Leman Akoglu. Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 985–994, 2015.

Amit Roy, Juan Shu, Jia Li, Carl Yang, Olivier Elshocht, Jeroen Smeets, and Pan Li. GAD-NR: Graph anomaly detection via neighborhood reconstruction. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pp. 576–585, 2024.

Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *The Semantic Web*, pp. 593–607, 2018.

Fengzhao Shi, Yanan Cao, Yanmin Shang, Yuchen Zhou, Chuan Zhou, and Jia Wu. H2-FDetector: A gnn-based fraud detector with homophilic and heterophilic connections. In *Proceedings of the ACM Web Conference 2022*, pp. 1486–1494, 2022.

Weihu Song, Lei Li, Mengxiao Zhu, Yue Pei, and Haogang Zhu. MMP: Enhancing unsupervised graph anomaly detection with multi-view message passing. *Pattern Recognition*, 172:112388, 2026.

Jiangnan Tang, Huanhuan Gu, Darko B. Vuković, Guandong Xu, Youquan Wang, Haicheng Tao, and Jie Cao. Fraud detection in multi-relation graph: Contrastive learning on feature and structural levels. *Neurocomputing*, 637:130063, 2025.

Jianheng Tang, Fengrui Hua, Ziqi Gao, Peilin Zhao, and Jia Li. GADBench: Revisiting and benchmarking supervised graph anomaly detection. In *Advances in Neural Information Processing Systems*, volume 36, pp. 29628–29653, 2023.

Ruidong Wang, Liang Xi, Fengbin Zhang, Haoyi Fan, Xu Yu, Lei Liu, Shui Yu, and Victor C. M. Leung. Context correlation discrepancy analysis for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 37(1):174–187, 2025a.

Shihui Wang, Yulan Yang, Zixin Tan, Zhigao Zheng, Jiawei Jiang, Hao Huang, Quanqing Xu, and Chuanhui Yang. GADACE: Graph anomaly detection combining attribute contrast and structure reconstruction. In *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2025b.

Xuhong Wang, Baihong Jin, Ying Du, Ping Cui, Yingshui Tan, and Yupu Yang. One-class graph neural networks for anomaly detection in attributed networks. *Neural Computing and Applications*, 33(18):12073–12085, 2021.

Cong Wu, Jianfei Sun, Jing Chen, Mamoun Alazab, Yang Liu, and Yang Xiang. TCG-IDS : Robust network intrusion detection via temporal contrastive graph learning. *IEEE Transactions on Information Forensics and Security*, 20:1475–1486, 2025.

Sheng Xiang, Mingzhi Zhu, Dawei Cheng, Enxia Li, Ruihui Zhao, Yi Ouyang, Ling Chen, and Yefeng Zheng. Semi-supervised credit card fraud detection via attribute-driven graph representation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(12):14557–14565, 2023.

Yiming Xu, Zhen Peng, Bin Shi, Xu Hua, Bo Dong, Song Wang, and Chen Chen. Revisiting graph contrastive learning on anomaly detection: A structural imbalance perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(12):12972–12980, 2025.

Huiliang Zhai, Xiangyi Teng, and Jing Liu. SFGA: Similarity-constrained fusion learning for unsupervised anomaly detection in multiplex graphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(33):28140–28149, 2026.

Ge Zhang, Jia Wu, Jian Yang, Amin Beheshti, Shan Xue, Chuan Zhou, and Quan Z. Sheng. FRAUDRE: Fraud detection dual-resistant to graph inconsistency and imbalance. In *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 867–876, 2021.

Xudong Zhao, Xiaolong Xu, Haolong Xiang, Tong Gao, Lianyong Qi, Amin Beheshti, Xuyun Zhang, and Wanchun Dou. DyLogNet: A dynamic multi-relational graph framework for log anomaly detection. In *Proceedings of the ACM Web Conference 2026*, pp. 3755–3763, 2026.

Yunfeng Zhao, Yixin Liu, Shiyuan Li, Qingfeng Chen, Yu Zheng, and Shirui Pan. FreeGAD: A training-free yet effective approach for graph anomaly detection. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 4379–4389, 2025.

Yu Zheng, Ming Jin, Yixin Liu, Lianhua Chi, Khoa T. Phan, and Yi-Ping Phoebe Chen. Generative and contrastive self-supervised learning for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12220–12233, 2023.

A NOTATION

Table 4 summarizes the key notation used throughout this paper. In general, calligraphic uppercase letters denote graphs or sets, boldface uppercase letters denote matrices or tensors, boldface lowercase letters denote vectors, and italic letters denote scalars or indices. Multi-letter descriptive subscripts and superscripts are typeset in upright Roman font.

Table 4: Summary of key notation.

Symbol	Description
$\mathcal{G} = \{\mathcal{G}^{(1)}, \ldots, \mathcal{G}^{(R)}\}$	Multi-relation graph with R relation types
$\mathcal{V}$, N	Shared node set and number of nodes
$\mathcal{E}^{(r)}$, $\mathbf{A}^{(r)}$	Edge set and adjacency matrix of relation r
$\mathbf{X}$	Shared node feature matrix
$\mathbf{Z}^{(r)}$	Node embeddings under relation r
$\mathbf{E}$	Relation-wise reconstruction residual matrix
$\mathbf{W}$	Final node-specific relation weight matrix
$\mathbf{D}$	Relation-wise degree matrix
$\mathcal{V}_{\mathrm{Ex}}$, $\mathcal{V}_{\mathrm{Im}}$	Explicit- and implicit-inconsistent node sets
s_i^{Ex}, s_i^{Im}	Explicit and implicit inconsistency scores of node v_i
$s_{\mathrm{incons},i}^{(t)}$	Inconsistency score of node v_i at epoch t
$\ell_{\mathrm{combined},i}$	Relation-weighted reconstruction objective of node v_i
ℓ_{recon}	Overall training objective
s_i	Final anomaly score of node v_i

B PRELIMINARY STUDY

Jaccard Distance. To quantify the disagreement between relation-specific anomaly sets, we adopt the Jaccard distance, defined for two sets $\mathcal{A}^{(r)}$ and $\mathcal{A}^{(r')}$ as

$$d_{\mathrm{J}}\left(\mathcal{A}^{(r)}, \mathcal{A}^{(r')}\right) = 1 - \frac{\left|\mathcal{A}^{(r)} \cap \mathcal{A}^{(r')}\right|}{\left|\mathcal{A}^{(r)} \cup \mathcal{A}^{(r')}\right|}, \tag{8}$$

where a value close to 1 indicates that the two sets share very few common nodes, and a value close to 0 indicates near-complete overlap. The same metric is applied to the normal sets $\mathcal{N}^{(r)}$ for comparison.

C METHODOLOGY COMPARISON WITH MULTI-RELATION GAD METHODS

RelIn-GAD differs from existing unsupervised multi-relation GAD methods in its inconsistency taxonomy, scoring mechanism, and plug-and-play integration, as summarized in Table 5.

Existing unsupervised multi-relation methods aggregate information across relations, yet they mainly serve dedicated architectures to learn relation-wise representations under reconstruction, rather than explicitly modeling cross-relation inconsistency from separate views. Their anomaly scores are primarily derived from relation-wise reconstruction errors, optionally combined with additional intra-relation anomaly cues:

- AnomMAN (Chen et al., 2023) applies view-level attention to fuse relation-specific embeddings learned by multiple autoencoders. It models attribute and structural reconstruction errors within each relation as anomaly signals and aggregates the relation-wise reconstruction errors to obtain the final anomaly score.
- UMGAD (Li et al., 2025b) captures relation correlations through global relation weighting, graph augmentation, multi-view reconstruction, and contrastive learning. Its anomaly score is obtained by aggregating relation-wise reconstruction errors and additionally incorporating augmented graph views.

- SFGA (Zhai et al., 2026) performs cross-relation interaction through attention and similarity constraints within a relation-aware autoencoder. Cross-relation inconsistency is only implicitly captured through attention-based fusion of relation-specific encoded representations. Its anomaly score combines relation-wise reconstruction errors with intra-relation local dissimilarities between nodes and their neighbors.

Table 5: Comparison of cross-relation modeling and scoring mechanisms in representative unsupervised multi-relation GAD methods.

Method	Cross-relation Formulation	Inconsistency Taxonomy	Scoring Mechanism	Plug-and-play Integration
AnomMAN	View-level attention	No explicit modeling of cross-relation inconsistency	Attention-integrated reconstruction scoring	×
UMGAD	Relation weighting & multi-view learning	No explicit modeling of cross-relation inconsistency	Aggregation of relation- and view-wise reconstruction errors	×
SFGA	Relation weighting in a focal graph	Implicit cross-relation inconsistency from distinct encoders	Intra-relation local dissimilarities & relation-specific reconstruction errors	×
RelIn-GAD (Ours)	Global relation prior & node-specific refinement	Explicit structural and implicit reconstruction inconsistencies from a shared encoder	Reliability-weighted evidence aggregation & inconsistency-guided score correction	✓

RelIn-GAD differs conceptually, rather than merely in terminology or implementation, in the following two aspects: First, *RelIn-GAD introduces an explicit taxonomy of complementary cross-relation inconsistencies.* Explicit structural inconsistency is derived directly from deviations in relation-specific connectivity patterns and is independent of the detector. Implicit reconstruction inconsistency is derived from disagreement among relation-wise reconstruction outputs and therefore captures detector-level evidence. Existing methods use cross-relation information primarily for representation fusion, reconstruction, or consistency regularization, without explicitly separating structural and reconstruction-output inconsistencies. Second, *RelIn-GAD uses these inconsistencies differently in anomaly scoring.* Rather than embedding them only within representation learning, RelIn-GAD uses them to estimate global-to-local relation reliability, calibrate relation-wise reconstruction evidence, and correct the final anomaly score produced by the backbone.

D Additional Analysis of Adaptive Relation Weighting

We further compare RelIn-GAD with a uniform-weighting variant (Avg.), where all relations are assigned equal weights. As shown in Table 6, adaptively modeling cross-relation variation consistently improves anomaly detection performance across all four datasets.

Table 6: Effectiveness comparison of uniform (Avg.) and adaptive relation weighting (RelIn) under one fixed random seed. Each entry reports AUROC / AUPRC (%).

Method	Amazon	YelpChi	S-FFSD	IEEE-CIS
Avg.	86.03 / 26.12	61.25 / 23.33	62.71 / 25.28	67.96 / 16.16
RelIn	88.53 / 35.58	63.68 / 25.23	63.97 / 27.96	68.00 / 16.47

E Analysis of the Node-Adaptive Objective

E.1 Effect of Inconsistency-Aware Scaling

We further analyze how the inconsistency-aware scaling in the node-adaptive training objective of Eq. (6) affects optimization. Let

$$u_i = \ell_{\text{combined},i} \geq 0, \qquad \sigma_i = \sigma_i^{(t)}$$

for simplicity. In the implementation, the inconsistency score is detached before computing the scale

$$\sigma_i = 1 + \mathrm{sg}\Big(s^{(t)}_{\mathrm{incons},i}\Big)\,(\sigma_{\max} - 1) \in [1, \sigma_{\max}],$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator. Thus, σ_i is treated as a fixed node-wise coefficient during backpropagation, and gradients do not propagate through $s^{(t)}_{\mathrm{incons},i}$.

The node-level objective is

$$\ell_{\mathrm{recon},i} = \mathrm{softplus}(\sigma_i u_i).$$

Its derivative with respect to the relation-weighted reconstruction loss is

$$\frac{\partial \ell_{\mathrm{recon},i}}{\partial u_i} = \sigma_i \,\mathrm{sigmoid}(\sigma_i u_i). \tag{9}$$

Because $u_i \geq 0$ and $\sigma_i \geq 1$,

$$\frac{\sigma_i}{2} \leq \sigma_i \,\mathrm{sigmoid}(\sigma_i u_i) < \sigma_i. \tag{10}$$

Accordingly, for model parameters Θ that affect u_i,

$$\frac{\partial \ell_{\mathrm{recon},i}}{\partial \Theta} = \underbrace{\sigma_i \,\mathrm{sigmoid}(\sigma_i u_i)}_{\text{node-adaptive gradient multiplier}} \frac{\partial u_i}{\partial \Theta}. \tag{11}$$

Therefore, for a fixed non-negative u_i, increasing $s^{(t)}_{\mathrm{incons},i}$ increases σ_i and strengthens the node's gradient contribution. The sigmoid factor additionally depends on the current reconstruction loss and approaches one as $\sigma_i u_i$ increases.

For relation-consistent nodes, $s^{(t)}_{\mathrm{incons},i} = 0$ gives $\sigma_i = 1$. Such nodes receive no inconsistency-dependent scaling and are optimized using the unscaled softplus transformation. This adaptive transformation is applied only during training; inference uses the anomaly score defined in Eq. (7).

E.2 Rationale for Explicit Structural Correction

The attention-weighted reconstruction term in Eq. (7) captures relation-specific reconstruction evidence, but reconstruction alone may not fully reflect pronounced structural disparity across relations. In particular, when a node has very limited connectivity under one relation, the corresponding reconstruction residual may provide weak evidence about its cross-relation structural behavior.

Explicit inconsistency provides a complementary, reconstruction-independent signal through s_i^{Ex}, which measures relative degree variation across relations. We therefore add γs_i^{Ex} to the anomaly score only when $v_i \in \mathcal{V}_{\mathrm{Ex}}$. This gated design supplements reconstruction evidence for nodes exhibiting the high–low connectivity contrast specified in Definition 1, while avoiding the indiscriminate use of structural variation as an anomaly criterion.

The correction is not applied to relation-consistent nodes. Although such nodes may exhibit low connectivity under some relations, they do not satisfy the explicit-inconsistency condition requiring sufficiently high connectivity under another relation. Applying the structural correction to these nodes could therefore introduce degree variation that does not correspond to the explicit cross-relation contrast targeted by RelIn-GAD.

F Case Study

To illustrate the two forms of cross-relation inconsistency captured by RelIn-GAD, we examine two labeled anomalous nodes from Amazon as post-hoc case studies: v_{6821} for explicit inconsistency and v_{7769} for implicit inconsistency. Ground-truth labels are used only for visualization and interpretation and are never accessed by RelIn-GAD. Fig. 3 visualizes their first-order ego-neighborhoods under the three relations. For Amazon, the default thresholds are $\theta_{\mathrm{low}} = 0$ and $\theta_{\mathrm{high}} = 629.85$.

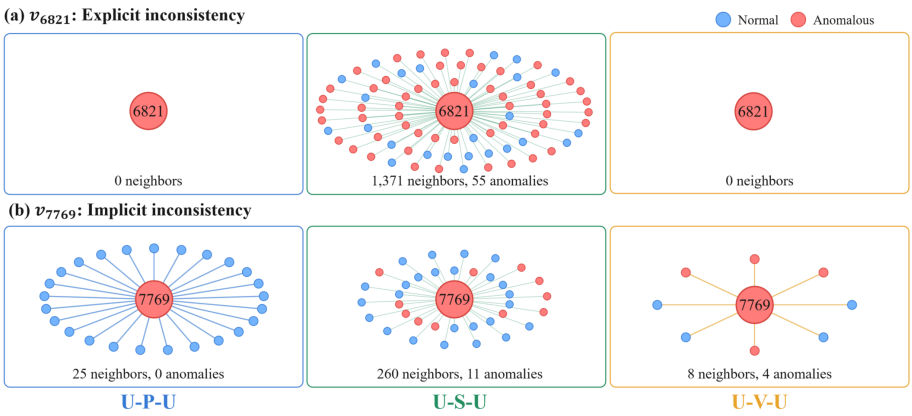


Figure 3: Relation-specific first-order ego-neighbourhoods of two labelled Amazon anomalies: (a) Node v_{6821} exhibits explicit inconsistency; its learned weights for U-P-U, U-S-U, and U-V-U are 0.148, 0.258, and 0.594, respectively. (b) Node v_{7769} exhibits implicit inconsistency; its corresponding relation weights are 0.167, 0.816, and 0.017, respectively. In the two U-S-U views, all anomalous neighbours and 25 sampled normal neighbours are shown; the other views show complete neighbourhoods.

Explicit inconsistency. Node v_{6821} has relation-wise degrees $(0, 1371, 0)$ under U-P-U, U-S-U, and U-V-U, respectively. It is isolated under U-P-U and U-V-U but densely connected under U-S-U, thereby satisfying the explicit-inconsistency criterion. Moreover, only 55 of its $1{,}371$ U-S-U neighbors are labeled anomalous (4.0%), indicating that the salient evidence lies in the sharp cross-relation connectivity contrast rather than in an anomalous majority within a single relation.

Its reconstruction behavior further illustrates why structural inconsistency provides complementary information beyond reconstruction alone. The structurally empty U-V-U relation yields the smallest reconstruction residual, yet receives the largest learned relation weight (0.594). Thus, the pronounced high–low connectivity contrast is not necessarily reflected by relation-weighted reconstruction evidence. Accordingly, the explicit inconsistency score contributes to $s^{(t)}_{\mathrm{incons},i}$ through Eq. (4), modulates the node-adaptive training objective in Eq. (6), and additionally provides the gated structural correction in Eq. (7) at inference.

Implicit inconsistency. Node v_{7769} remains connected under all three relations, with relation-wise degrees $(25, 260, 8)$, and therefore belongs to $\mathcal{V}_{\mathrm{Im}}$. Its labeled neighborhood composition also varies markedly across relations: no anomalous neighbors appear under U-P-U, only a small fraction appear under U-S-U, whereas half of the U-V-U neighbors are anomalous.

More importantly, the U-V-U relation exhibits a noticeably larger reconstruction residual than the other two relations, revealing clear disagreement in relation-specific reconstruction behavior as defined in Eq. (1). At the same time, the learned relation weights for U-P-U, U-S-U, and U-V-U are 0.167, 0.816, and 0.017, respectively, greatly reducing the direct contribution of U-V-U to the relation-weighted reconstruction term in Eq. (5). Nevertheless, its residual remains part of the implicit inconsistency score s^{Im}_i in Eq. (3). Through Eq. (4), this cross-relation reconstruction disagreement contributes to $s^{(t)}_{\mathrm{incons},i}$ and consequently modulates the node-adaptive training objective in Eq. (6). Because $v_{7769} \notin \mathcal{V}_{\mathrm{Ex}}$, no structural correction is applied in Eq. (7) at inference.

Together, these two cases highlight the complementary roles of the two inconsistency signals. Explicit inconsistency captures pronounced structural disparity across relations, whereas implicit inconsistency captures disagreement in relation-specific reconstruction behavior. Both signals contribute to inconsistency-aware training modulation, while only explicit inconsistency provides the gated structural correction at inference. Neighbor labels are shown solely for post-hoc interpretation and are not used to compute the inconsistency scores or train RelIn-GAD.

G SENSITIVITY ANALYSES

G.1 ROBUSTNESS TO THE NUMBER AND SEMANTIC QUALITY OF RELATIONS

We further evaluate sensitivity to the number and semantic quality of available relation types. We use each relation's single-relation reconstruction AUROC as a proxy for its semantic quality, rank relations in descending order by this measure, and then add them incrementally. Table 7 reports the resulting AUROC. On Amazon, performance improves steadily as additional relations are included. On S-FFSD, performance remains stable and reaches its highest value when all four relations are used. This suggests that RelIn-GAD can benefit from complementary informative relations while limiting the adverse effect of less informative ones.

Table 7: AUROC (%) as relations are incrementally added in descending order of single-relation reconstruction quality under one fixed random seed.

#Relations	Amazon	S-FFSD
1	83.87	54.14
2	87.30	53.86
3	88.70	54.45
4	–	57.21

G.2 HYPERPARAMETER ANALYSIS

We analyze the sensitivity of RelIn-GAD with backbone model AnomalyDAE to six key hyperparameters: the reconstruction balance coefficient α, the DI structural correction weight γ, the embedding dimension q, the maximum softplus scale $\sigma_{\max}$, the minimum and initial attention temperatures $\tau_{\min}$ and τ_0. As shown in Fig. 4, overall, the model demonstrates stable performance across a broad range of parameter values on most datasets. For the α, Amazon is sensitive in the lower range, peaking around $\alpha \in [0.4, 0.6]$, while the other three datasets remain relatively flat throughout, suggesting $\alpha \in [0.4, 0.6]$. The γ has minimal impact on all datasets except Amazon, where performance peaks at $[0.1, 0.3]$. For the q, all datasets show low sensitivity across the tested range $[8, 64]$. Regarding the $\sigma_{\max}$, Amazon benefits from larger values with performance improving steadily beyond $\sigma_{\max} = 1.5$, while other datasets remain stable. For the attention temperatures, $\tau_{\min} \in [0.4, 0.6]$ yields consistent performance across datasets, and τ_0 is robust over a wide range $[2.2, 4.7]$ with no sharp transitions observed. These results suggest that RelIn-GAD is generally robust to hyperparameter choices, requiring careful tuning primarily on datasets with strong relational inconsistency signals such as Amazon.

Degree Thresholds and Annealing Schedule. We additionally analyze the degree thresholds and the annealing schedule on Amazon. The degree-threshold results are reported in Table 8. The default values are $\theta_{\text{low}} = 0$ and $\theta_{\text{high}} = \frac{1}{N}\sum_{i=1}^{N} \max_r d_i^{(r)}$. Let $\theta_{\text{high}}^{\text{def}}$ denote this default value of θ_{high}. The AUROC range is only 0.12 percentage points for θ_{low} and 1.10 percentage points for θ_{high}, indicating low sensitivity to the degree-threshold choices.

Table 8: Sensitivity to the degree thresholds on Amazon under one fixed random seed.

θ_{low}	AUROC (%)	θ_{high}	AUROC (%)
0	88.70	$0.50 \cdot \theta_{\text{high}}^{\text{def}}$	88.37
1	88.58	$0.75 \cdot \theta_{\text{high}}^{\text{def}}$	88.97
2	88.58	$1.00 \cdot \theta_{\text{high}}^{\text{def}}$	88.70
3	88.64	$1.25 \cdot \theta_{\text{high}}^{\text{def}}$	87.87
4	88.64	$1.50 \cdot \theta_{\text{high}}^{\text{def}}$	87.87

The annealed coefficient is defined in Eq. (4) through $\beta_t = \beta_{\text{end}} + (\beta_{\text{start}} - \beta_{\text{end}})(1 - t/T)$, with default $\beta_{\text{start}} = 1.0$ and $\beta_{\text{end}} = 0.15$. Since β_t is used as a mixing coefficient, we restrict the

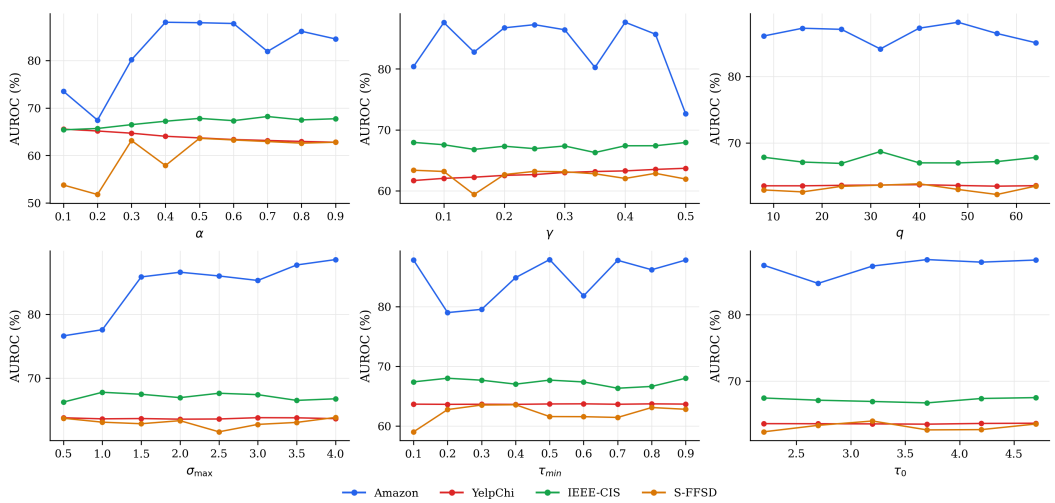


Figure 4: Hyperparameter sensitivity analysis.

sensitivity analysis of β_{start} to the valid interval $[0, 1]$. Table 9 reports the sensitivity to β_{start} and β_{end}.

Table 9: Sensitivity to the annealing parameters on Amazon under one fixed random seed.

β_{start}	**AUROC (%)**	β_{end}	**AUROC (%)**
0.50	85.48	0.0	88.58
0.75	84.03	0.1	88.65
1.00	88.70	0.2	88.69
–	–	0.3	88.73
–	–	0.4	88.68

The model is highly stable with respect to β_{end}, whose AUROC ranges from 88.58% to 88.73%. It is comparatively more sensitive to β_{start}: the best tested setting is the default $\beta_{\text{start}} = 1.0$, while smaller initial values can degrade performance. This agrees with the role of β_{start} in emphasizing the static structural signal during early training, when reconstruction errors are not yet well calibrated.

H Relation Weight Behavior under Controlled Relation Perturbations

To examine whether the learned relation weights respond meaningfully to changes in relation quality, we conduct controlled structural perturbations on the Amazon dataset. For highly correlated, missing, and redundant perturbations, we randomly select 5% of anomalous nodes; for the noisy perturbation, we select 10%. Ground-truth labels are used only to construct this post-hoc controlled perturbation benchmark and are never used for model training or relation-weight estimation. For each selected node, structural perturbations are applied to a perturbed relation indexed by r_{pert}, while a target relation indexed by r_{tar} serves as the reference for highly correlated and redundant perturbations. Let $d^{(r)}(v)$ and $\Gamma^{(r)}(v)$ denote the degree and neighbor set of node v under relation r, respectively. We construct four perturbation types:

- **Highly correlated:** replace the local edges of r_{pert} with those of r_{tar}, then add or remove $0.05\, d^{(r_{\text{tar}})}(v)$ random edges while maintaining a Jaccard similarity of at least 95% with r_{tar}.
- **Noisy:** add $0.10\, d^{(r_{\text{pert}})}(v)$ random edges that do not exist in r_{pert}.
- **Missing:** remove $0.05\, d^{(r_{\text{pert}})}(v)$ random edges from r_{pert}.
- **Redundant:** completely duplicate the local edges of r_{tar} into r_{pert}.

All comparisons are made between the final node-specific relation weights before and after perturbation. For a compared node v_i, let $W^{\mathrm{pre}}_{i,r_{\mathrm{pert}}}$ and $W^{\mathrm{post}}_{i,r_{\mathrm{pert}}}$ denote the final normalized weights assigned to the perturbed relation before and after perturbation, respectively, and define $W^{\mathrm{pre}}_{i,r_{\mathrm{tar}}}$ and $W^{\mathrm{post}}_{i,r_{\mathrm{tar}}}$ analogously for the target relation. We say that the perturbed relation's weight *shifts toward* the target relation's weight if

$$\left|W^{\mathrm{post}}_{i,r_{\mathrm{pert}}} - W^{\mathrm{post}}_{i,r_{\mathrm{tar}}}\right| < \left|W^{\mathrm{pre}}_{i,r_{\mathrm{pert}}} - W^{\mathrm{pre}}_{i,r_{\mathrm{tar}}}\right|, \tag{12}$$

and that the perturbed relation's weight *decreases* if

$$W^{\mathrm{post}}_{i,r_{\mathrm{pert}}} < W^{\mathrm{pre}}_{i,r_{\mathrm{pert}}}. \tag{13}$$

The percentages in Table 10 are computed over the compared nodes, i.e., as the number of nodes satisfying the corresponding condition divided by the total number of compared nodes.

Table 10: Percentage of perturbed nodes whose perturbed-relation weight either shifts toward the target-relation weight or decreases under controlled relation perturbations on Amazon.

Perturbation	**Shift toward target (%)**	**Weight decrease (%)**
Highly correlated	73.17	–
Noisy	–	62.20
Missing	–	60.97
Redundant	80.49	–

The learned weights respond consistently to controlled changes in relation quality. Under highly correlated and redundant perturbations, the perturbed relation tends to receive a weight closer to the target relation, whereas noisy and missing perturbations more often reduce the perturbed relation's weight. These results indicate that the relation-weighting mechanism adapts to structural similarity and degradation rather than assigning fixed importance to relation types.

I Details of Experimental Implementation

I.1 Introduction to Datasets

Amazon (McAuley & Leskovec, 2013) is a standard multi-relation fraud-detection benchmark derived from Amazon reviews in the Musical Instruments category. Each node represents a user, and the dataset provides three predefined relation types: U-P-U connects users who review at least one common product, U-S-U connects users exhibiting similar rating behavior, and U-V-U captures similarity in review text. Users with suspicious review behavior are treated as anomalies.

YelpChi (Rayana & Akoglu, 2015) is a standard multi-relation review-spam benchmark derived from Yelp reviews, where each node represents a review. We directly use the three predefined relations provided with the dataset: R-U-R connects reviews posted by the same user, R-T-R connects reviews of the same product posted in the same month, and R-S-R connects reviews of the same product with the same star rating. Filtered reviews are treated as anomalous, whereas recommended reviews are treated as normal.

S-FFSD (Xiang et al., 2023) is a simulated and scaled-down public version of the Financial Fraud Semi-supervised Dataset (FFSD), where each node represents a transaction. We preprocess S-FFSD to construct four relation views based on shared transaction attributes—*Source*, *Target*, *Location*, and *Type*—as detailed in Appendix I.3.

IEEE-CIS (Ge et al., 2020) is a real-world e-commerce transaction fraud dataset released through the IEEE-CIS Fraud Detection competition, with data provided by Vesta Corporation. Each node represents a transaction. We directly use the provided three relation views: T-C-T, T-L-T, and T-M-T, which group transactions according to shared `card1`, `addr1`, and `P_emaildomain` values, respectively. Fraudulent transactions are treated as anomalies.

I.2 Data Split

Due to the memory constraints of full-graph training in unsupervised graph anomaly detection, we adopt a node partitioning strategy for relatively large-scale datasets. Specifically, the node set $\mathcal{V}$ is divided into n non-overlapping subsets $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_n\}$ of equal size, where $\sum_{j=1}^{n} |\mathcal{V}_j| = N$ and $|\mathcal{V}_j| = N/n$. For each subset, the corresponding subgraph is constructed as an induced subgraph based on each relational adjacency matrix. This yields a series of smaller graphs used for model training. The number of nodes in each subgraph is 15,318 for YelpChi, 9,881 for S-FFSD, and 15,000 for IEEE-CIS, to preserve as much large graph structure information as possible.

Table 11: Statistics of the experimental datasets

Dataset	#Nodes	Relation	#Edges	Avg. Degree (Ano. / Nor.)	#Anomalies
Amazon	11,944	U-P-U	175,608	49.46 / 27.93	821
		U-S-U	3,566,479	195.44 / 626.85	
		U-V-U	1,036,737	42.39 / 183.28	
YelpChi	45,954	R-U-R	49,315	0.29 / 2.46	6,677
		R-T-R	573,616	25.10 / 24.94	
		R-S-R	3,402,743	142.57 / 149.03	
S-FFSD	29,643	T-L-T	58,633	3.99 / 3.95	5,256
		T-C-T	47,624	2.06 / 3.46	
		T-D-T	58,910	3.98 / 3.97	
		T-T-T	57,572	3.93 / 3.87	
IEEE-CIS	30,000	T-C-T	229,157	21.24 / 14.70	2,642
		T-L-T	194,308	13.06 / 12.94	
		T-M-T	188,883	4.96 / 13.33	

I.3 S-FFSD Dataset Preprocessing

The S-FFSD dataset contains transaction records annotated with three classes: normal (label 0), fraudulent (label 1), and an unlabeled (label 2) that is excluded from our experiments. After filtering, the dataset comprises 29,643 nodes, among which 17.73% are fraudulent.

Feature Construction. Node features are extracted from the S-FFSDneofull.csv file, which provides 126 attributes per transaction after removing the label column. Raw features are first enriched through time-window-based feature engineering: for each transaction, we compute statistical aggregates (mean, sum, standard deviation, and bias of transaction amounts, as well as the number of unique targets, locations, and types) over 15 time windows of varying lengths, capturing temporal behavioral patterns at multiple scales. Missing values introduced during this process are filled with zero. Categorical attributes (`Source`, `Target`, `Location`, `Type`) are then encoded via `LabelEncoder` before being included in the feature matrix. The resulting feature matrix is of shape $29{,}643 \times 126$.

Multi-relation Graph Construction. We construct four relations based on shared transaction attributes: *Source*, *Target*, *Location*, and *Type*. For each relation, transactions sharing the same attribute value are grouped together and sorted by timestamp. Within each group, we connect each transaction to its k subsequent transactions (with $k = 3$), yielding directed edges that capture temporal proximity among similar transactions. The directed edges are then symmetrized to obtain undirected adjacency matrices, with duplicate entries set to one. Each relation is stored as a sparse CSR matrix of shape $29{,}643 \times 29{,}643$. The total number of edges across all four relations gives an edge-to-node ratio of approximately 18.67, consistent with the statistics reported in the original dataset.

Dataset Format. The preprocessing pipeline produces node features, labels, and four sparse adjacency matrices corresponding to the *Source*, *Target*, *Location*, and *Type* relations, respectively. The preprocessed data are consolidated into a single `S-FFSD.mat` file using `scipy.io.savemat`. The file contains the following fields:

- `features`: the node feature matrix of shape $(29{,}643 \times 126)$;
- `label`: the node label vector of length $29{,}643$;
- `net_tlt`: the sparse adjacency matrix for the *Location* relation;
- `net_tct`: the sparse adjacency matrix for the *Source* relation;
- `net_tdt`: the sparse adjacency matrix for the *Type* relation;
- `net_ttt`: the sparse adjacency matrix for the *Target* relation.

At load time, the four relation matrices are retrieved as `relation_list` = [`net_tlt`, `net_tct`, `net_tdt`, `net_ttt`], following the same interface used by other datasets in our experiments.

I.4 Reconstruction-based Models Integration Details

To extend single-relation reconstruction-based detectors to the multi-relation setting, we replace each model's original training objective with our inconsistency-aware loss while keeping the encoder architectures intact. The integration follows a unified protocol applied consistently across all models.

Specifically, each encoder in the original model is shared across all R relations: given a relation r, the shared encoder and independent decoder process the graph under relation r to produce the corresponding node embedding representation and its associated reconstruction loss. Then passed to our **RelIn-GAD** module. The resulting inconsistency-aware loss replaces the original training objective, so that the gradients flowing back to the encoders are modulated by the inconsistency scores.

This protocol is applied uniformly even when a model comprises multiple sequential encoders. For instance, ADA-GAD (He et al., 2024) employs three encoders (node-level, subgraph-level, and edge-level), and DiffGAD (Li et al., 2025a) employs three components (a graph encoder, an unconditioned diffusion model, and a conditioned diffusion model); in both cases, each component is extended to operate over all R relations following the same procedure described above.

J Computational Complexity Analysis

The dominant cost of RelIn-GAD comes from three components. The shared GNN encoder processes each relation independently, giving a per-epoch cost of $\mathcal{O}(\sum_r |\mathcal{E}^{(r)}|q + NRq^2)$. The SE-Relation component performs a global pooling over $\hat{\mathbf{Z}} \in \mathbb{R}^{N\times R\times q}$ followed by two linear layers of sizes $R \to m \to R$, costing $\mathcal{O}(NRq + Rm)$, which is negligible relative to the encoder, where m denotes the hidden dimension of the node-attention MLP. The node-attention MLP takes input of dimension $R(q+1)$ and outputs R logits per node, costing $\mathcal{O}(N \cdot R(q+1) \cdot m)$. The classified inconsistency score requires only degree statistics and per-relation reconstruction errors, both of which are $\mathcal{O}(NR)$. The overall per-epoch complexity is therefore $\mathcal{O}(\sum_r |\mathcal{E}^{(r)}|q + NRq^2)$, linear in the number of edges and nodes, matching the complexity of standard single-relation GNN anomaly detectors up to a factor of R in the node-wise term.

J.1 Empirical Runtime and Memory

Comparison with Unsupervised Multi-relation Methods. Beyond the theoretical analysis above, we further empirically evaluate the computational efficiency of RelIn-GAD against two unsupervised multi-relation baselines on Amazon and S-FFSD, measuring runtime and peak GPU memory over 100 training epochs, as reported in Table 12. The two datasets provide complementary settings: Amazon contains three relation types and a larger number of edges, whereas S-FFSD contains four relation types but fewer edges.

RelIn-GAD introduces moderate computational and memory overhead relative to the fastest baseline while remaining substantially faster than SFGA in these experiments. Its memory use is influenced by the number of relations because relation-specific embeddings and intermediate reconstruction outputs are retained for joint relation-reliability and inconsistency estimation.

Table 12: Runtime and peak GPU memory comparison on Amazon and S-FFSD over 100 training epochs under a fixed random seed.

Method	Amazon		S-FFSD	
	Runtime (s)	Memory (GB)	Runtime (s)	Memory (GB)
UMGAD (Li et al., 2025b)	59.20	4.24	5.38	3.04
SFGA (Zhai et al., 2026)	544.86	1.80	480.50	1.78
RelIn-GAD	128.27	2.26	12.24	4.24

Scalability with the Number of Relations. We also vary the number of active relation types and report the runtime and peak GPU memory of RelIn-GAD in Table 13. Peak memory increases monotonically with the number of relations on both datasets. Runtime generally increases as more relation types are introduced, while also depending on the size of each relation-specific graph. In particular, S-FFSD exhibits an approximately linear increase in runtime from one to four relations.

Table 13: Runtime and peak GPU memory of RelIn-GAD with increasing numbers of active relation types.

#Relations	Amazon		S-FFSD	
	Runtime (s)	Memory (GB)	Runtime (s)	Memory (GB)
1	5.64	0.74	4.00	1.93
2	99.14	1.97	6.71	2.70
3	128.27	2.26	9.48	3.47
4	–	–	12.24	4.24

K Social Impacts

From a positive perspective, the proposed method can bring benefits to applications such as fraud detection, network intrusion detection, and medical diagnosis. In these applications, identifying abnormal patterns has significant social value. On the negative side, like any anomaly detection system, this method may generate false alarms, which may mistakenly label legitimate entities as anomalies. This raises concerns about fairness and improper use in sensitive decision-making environments.