HYPERROGMAE: MULTI-SCALE SEMANTIC HYPERGRAPHS FOR ROBUST GRAPH MASKED AUTOENCODER AGAINST STRUCTURAL ATTACKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Graph Masked AutoEncoders (GMAEs) propagate contextual information through the observed pairwise topology to reconstruct masked graph elements. This paradigm makes them vulnerable to structural attacks that corrupt the observed topology. Existing robust GMAEs mitigate this vulnerability through adversarial masking, structural augmentation, and edge-level filtering or reweighting. However, their contextual modeling relies primarily on pairwise relations and thus lacks multi-scale, higher-order semantic context for robust reconstruction under structural attacks. To this end, we propose ***HyperRoGMAE***, a ***Hyper***graph-guided ***Ro***bust ***G***raph ***M***asked ***A***uto***E***ncoder that constructs a multi-scale semantic hypergraph for masked representation learning. Specifically, HyperRoGMAE first constructs a multi-scale semantic hypergraph and estimates semantic reliability based on assignment margins and cross-view stability. It then pretrains a hypergraph teacher that encodes masked views and provides reliability-weighted higher-order guidance. During student pretraining, it attenuates suspicious edges according to feature similarity and hypergraph-membership consistency. Finally, it introduces a node-aware adaptive fusion strategy to fuse graph and semantic representations and employs dual reconstruction objectives to preserve their complementary information. Extensive experiments on five datasets demonstrate the effectiveness of the proposed HyperRoGMAE under four structural attacks.

1 INTRODUCTION

Graph Masked AutoEncoders (GMAEs) have emerged as a prominent generative paradigm for graph representation learning (Hou et al., 2022). By masking node attributes or structures and reconstructing them from visible context, GMAEs learn informative node representations (Hou et al., 2023). Recent methods have advanced this paradigm via improved masking strategies and reconstruction objectives. For example, AUGMAE (Wang et al., 2024a) connects masked reconstruction with context-level contrastive learning and introduces easy-to-hard adversarial masking and uniformity regularization. ProtoMGAE (Zheng & Jia, 2024) combines masked feature reconstruction with momentum-based learning and prototype-aware clustering consistency to capture information at multiple levels. DGMAE (Zheng et al., 2025) reconstructs discrepancies between connected nodes to preserve their distinctions on heterophilic graphs. Despite these advances, contextual information from the observed topology can become unreliable under structural attacks.

Several robust GMAEs incorporate adversarial learning or structural refinement to improve reconstruction and representation learning. ArmorGAE (Zhang et al., 2026) proposes an adversarial masked graph modeling framework for robust graph autoencoding by designing dual-consistency topology augmentation and adversarial edge reconstruction. AMGAE (Hu et al., 2025) employs a GNN discriminator to distinguish reconstructed from original node features and improve generation quality. More directly targeting adversarial perturbations, SP-AGCL (In et al., 2023) constructs a feature-induced k-NN graph as a node similarity-preserving view and contrasts it with stochastic and adversarial graph views to improve robustness against structural attacks. These advances motivate a question: ***how can attribute-derived context support masked reconstruction when observed neighborhoods become unreliable?*** Beyond individual pairwise similarities, multi-scale overlapping hypergraphs offer a way to aggregate shared attribute context across different granularities. It

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

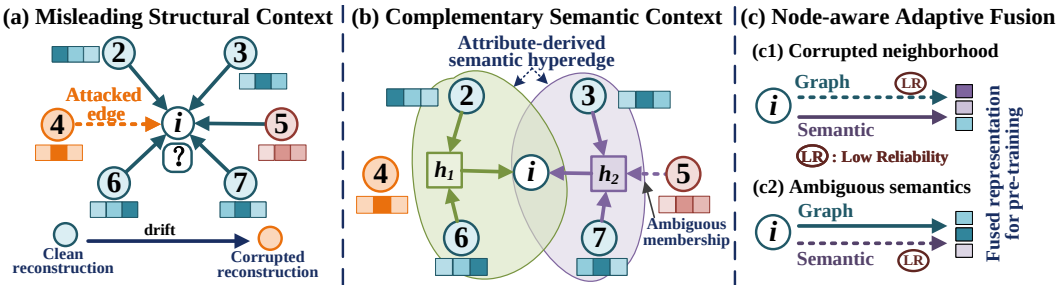


Figure 1: Motivation for HyperRoGMAE. (a) Structural attacks corrupt the observed topology. (b) Attribute-derived semantic hyperedges provide higher-order context. (c) Node-aware adaptive representation fusion regulates the contributions of graph and semantic representations.

provides an additional reference for reconstruction. However, leveraging highly reliable inferred relations and integrating observed graphs present two key challenges, as illustrated in Figure 1:

- **How can reliable semantic context be constructed?** Attribute-derived hypergraphs connect semantically related nodes beyond the observed neighborhood, providing complementary context for masked reconstruction. However, inferred memberships can be ambiguous, and a single hypergraph resolution may overlook relationships at other granularities. The challenge is to construct multi-scale context while accounting for membership uncertainty so that unreliable associations do not provide misleading guidance.
- **How can graph and semantic representations be adaptively integrated?** Even under structural attacks, the observed graph often retains valuable relational patterns that attribute-derived groups cannot capture. Conversely, semantic context carries its own uncertainty. Completely replacing graph topology or combining both modalities with fixed weights risks discarding valid structure or propagating noisy signals. Therefore, the challenge lies in integrating graph and semantic representations based on node-level reliability and cross-modal disagreement.

To this end, we propose ***HyperRoGMAE***, a ***Hyper***graph-guided ***Ro***bust ***G***raph ***M***asked ***A***uto***E***ncoder. Specifically, we first construct a multi-scale overlapping semantic hypergraph and estimate semantic reliability according to assignment margins and cross-view stability. We then pretrain and freeze a hypergraph teacher to provide reliability-weighted higher-order guidance. During student pretraining, the teacher encoder derives contextual hyperedge prompts from masked inputs. To retain useful graph information, the student graph encoder assigns continuous reliability weights to observed edges based on feature similarity and hypergraph-membership consistency. Furthermore, we propose a node-aware adaptive fusion strategy to integrate graph and semantic representations. Finally, we minimize the dual reconstruction loss to preserve complementary information from both branches. Our main contributions are summarized as follows:

- We propose HyperRoGMAE, which constructs an attribute-induced multi-scale semantic hypergraph and estimates semantic reliability to provide an explicit higher-order reference.
- We design a reliability-weighted graph student encoder to attenuate suspicious observed edges and a node-aware adaptive fusion strategy to transfer reliable semantic guidance.
- We conduct extensive experiments on five datasets whose results demonstrate the effectiveness and robustness of the proposed HyperRoGMAE under four structural attacks.

2 Related Work

Graph Masked Autoencoders and Robust Graph Learning. Graph masked autoencoders learn representations via the mask-reconstruction paradigm. GraphMAE (Hou et al., 2022) focuses on masked attribute reconstruction, while GraphMAE2 (Hou et al., 2023) strengthens this method via multi-view random re-mask decoding and latent representation prediction. Beyond attribute reconstruction, MaskGAE (Li et al., 2023) studies masked structure reconstruction, and Bandana

introduces continuous bandwidth masking and layer-wise bandwidth prediction to learn structural representations (Zhao et al., 2024). Robust graph learning addresses the influence of adversarial perturbations through various strategies (Wu et al., 2019; Zhang & Zitnik, 2020; Zhu et al., 2019; Jin et al., 2020). These defenses highlight the importance of controlling unreliable message passing. Self-supervised learning methods further explore adversarial contrastive views (Feng et al., 2024), feature-similarity preservation (In et al., 2023), and unsupervised structure refinement (Li et al., 2022). ArmorGAE (Zhang et al., 2026) employs dual-consistency topology augmentation and adversarial edge reconstruction. Despite their progress, these methods remain confined to pairwise topology and fail to harness higher-order semantic context for robust reconstruction.

Higher-order and Hypergraph Representation Learning. Hypergraph Neural Networks (HNNs) model higher-order dependencies by propagating the node-hyperedge incidence structures. Early methods generalized graph convolutions to hypergraphs or approximated hyperedges via pairwise expansions (Feng et al., 2019; Yadati et al., 2019). Subsequent models developed more unified message passing formulations and learnable multiset aggregation functions (Chien et al., 2022). These methods primarily study representation learning on a given hypergraph in supervised or semi-supervised settings. Self-supervised hypergraph learning further reduces label dependence by constructing pretext tasks from hypergraph structure and attributes. TriCL (Lee & Shin, 2023) performs contrastive learning at the node, hyperedge, and incidence levels, while HyperGCL (Wei et al., 2022) constructs contrastive views through fabricated and learnable generative augmentations. From a generative perspective, HypeBoy (Kim et al., 2024) introduces hyperedge filling as a self-supervised task for learning hypergraph representations. Additionally, HOG-Diff (Huang & Birdal, 2026) utilizes higher-order information to guide coarse-to-fine graph generation. However, they are mainly tailored for hypergraph inputs, leaving their potential for guiding robust masked autoencoding on attributed graphs under structural attacks unexplored.

3 METHODOLOGY

3.1 PRELIMINARIES

Definition 1: Attributed Graph (Zhao et al., 2023). Let $\mathcal{G}_o = (\mathcal{V}, \mathcal{E}_o, \mathbf{X})$ denote an observed attributed graph, where $\mathcal{V} = \{v_1, \ldots, v_N\}$ and $\mathcal{E}_o$ represent the sets of nodes and observed edges, and $\mathbf{X} \in \mathbb{R}^{N \times D}$ is the node attribute matrix. Its adjacency matrix is denoted by $\mathbf{A}_o \in \{0,1\}^{N \times N}$.

Definition 2: Graph Masked AutoEncoders (Hou et al., 2022). Given an adjacency matrix $\mathbf{A}_o$ and node attributes $\mathbf{X}$, GMAEs select a subset of nodes $\mathcal{V}_{\text{mask}} \subseteq \mathcal{V}$ and replace their attributes with a learnable mask token $\mathbf{x}_{\text{mask}}$, yielding a corrupted attribute matrix $\widetilde{\mathbf{X}}$. An encoder-decoder architecture $g_\phi(f_\theta(\widetilde{\mathbf{X}}, \mathbf{A}_o))$ is then trained to reconstruct the original attributes as $\widehat{\mathbf{X}}$. The reconstruction loss is computed on the masked set using the scaled cosine error (SCE) loss:

$$\mathcal{L}_{\text{sce}}(\widehat{\mathbf{X}}, \mathbf{X}; \mathcal{V}_{\text{mask}}) = \frac{1}{|\mathcal{V}_{\text{mask}}|} \sum_{i \in \mathcal{V}_{\text{mask}}} \left(1 - \frac{\widehat{\mathbf{x}}_i^\top \mathbf{x}_i}{\|\widehat{\mathbf{x}}_i\|_2 \|\mathbf{x}_i\|_2}\right)^\alpha, \alpha \geq 1. \tag{1}$$

where $\mathbf{x}_i$ and $\widehat{\mathbf{x}}_i$ denote the original and reconstructed attributes of node v_i, respectively.

Definition 3: *Attribute-induced Semantic Hypergraph*. Given a node set $\mathcal{V}$ with an attribute matrix $\mathbf{X} \in \mathbb{R}^{N \times D}$, an *attribute-induced semantic hypergraph* is formulated as $\mathcal{G}_s = (\mathcal{V}, \mathcal{E}_s)$, where $\mathcal{E}_s = \{e_1, \ldots, e_M\}$ denotes a set of semantic hyperedges inferred from $\mathbf{X}$. The higher-order structure of $\mathcal{G}_s$ is characterized by a weighted incidence matrix $\mathbf{H} \in \mathbb{R}^{N \times M}$, where $H_{im} > 0$ quantifies the soft membership strength of node v_i to hyperedge e_m.

Definition 4: *Semantic Hypergraph Propagation.* Given a weighted incidence matrix $\mathbf{H} \in \mathbb{R}^{N \times M}$, we define the node- and hyperedge-degree matrices as $\mathbf{D}_v = \text{diag}(\mathbf{H}\mathbf{1}_M)$ and $\mathbf{D}_e = \text{diag}(\mathbf{H}^\top \mathbf{1}_N)$. For a node representation matrix $\mathbf{Z} \in \mathbb{R}^{N \times d}$, semantic hypergraph propagation on $\mathcal{G}_s$ is defined as:

$$\mathcal{P}_H(\mathbf{Z}) = \mathbf{D}_v^{-1} \mathbf{H} \mathbf{D}_e^{-1} \mathbf{H}^\top \mathbf{Z}. \tag{2}$$

It aggregates node representations into hyperedges and then propagates the resulting group representations back to nodes. For fixed $\mathbf{H}$, this operator is equivalent to weighted node-to-node propagation with $\mathbf{A}_H = \mathbf{D}_v^{-1} \mathbf{H} \mathbf{D}_e^{-1} \mathbf{H}^\top$.

Problem: Robust Graph Representation Learning. Given an attributed graph $\mathcal{G}_o = (\mathcal{V}, \mathcal{E}_o, \mathbf{X})$, and an observed adjacency matrix $\mathbf{A}_o \in \{0,1\}^{N\times N}$, $\mathbf{A}_o$ may be corrupted by structural attacks. The objective is to learn a graph encoder f_θ that maps the input graph into low-dimensional node representations: $\mathbf{Z} = f_\theta(\mathbf{X}, \mathbf{A}_o)$. The learned embeddings $\mathbf{Z}$ preserve discriminative semantics for downstream tasks on both clean graphs and graphs under diverse structural attacks.

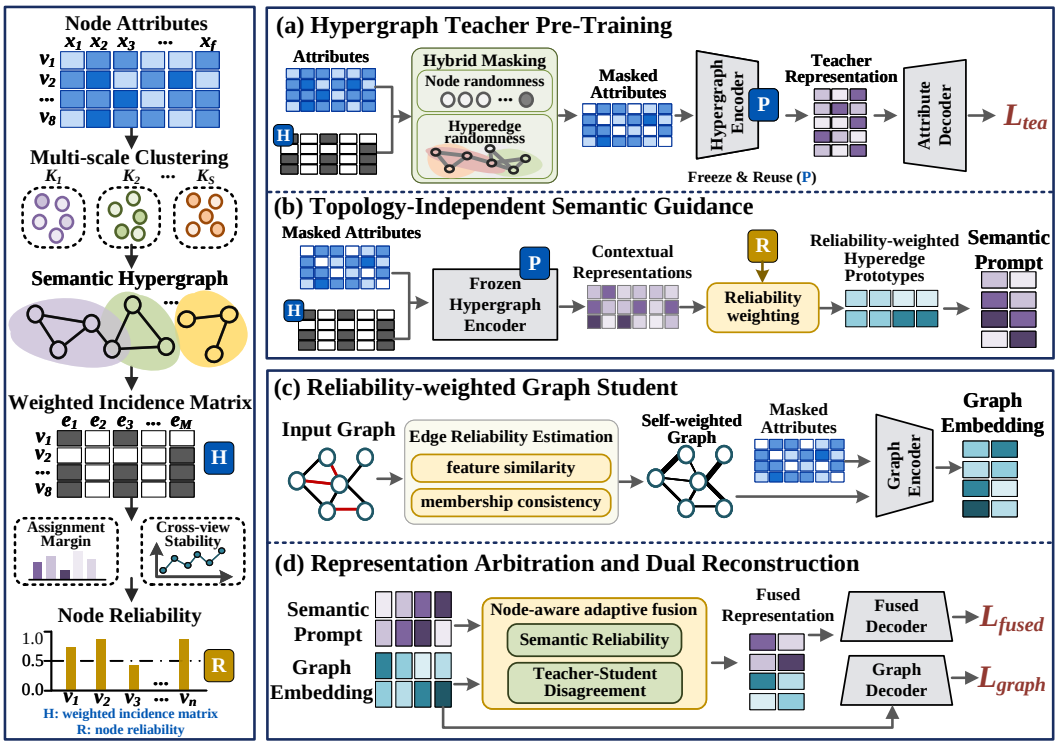


Figure 2: The framework of the proposed HyperRoGMAE.

3.2 Framework

In this subsection, we present HyperRoGMAE, a hypergraph-guided robust graph masked autoencoder. As illustrated in Figure 2, HyperRoGMAE first constructs a semantic hypergraph and estimates node reliability. Next, it pretrains a graph-independent hypergraph teacher to provide a stable higher-order semantic reference. It then trains a student encoder that estimates edge reliability and performs a soft reliability-weighted message passing. Finally, it designs a dual masked-attribute reconstruction loss to learn robust node representations. Additional preprocessing and gradient-flow details are provided in Appendix A.2.

3.3 Attribute-Induced Semantic Reference

Multi-scale Overlapping Semantic Hypergraph. To construct the attribute-induced semantic hypergraph, we first map node attributes $\mathbf{X}$ into a semantic space and apply row-wise ℓ_2 normalization to obtain the latent embedding matrix $\mathbf{Y} \in \mathbb{R}^{N\times d'}$. We then perform clustering at multiple semantic resolutions $\mathcal{K} = \{K_1, \ldots, K_S\}$. At each scale $s \in \{1, \ldots, S\}$, mini-batch k-means produces K_s cluster centers $\{\mathbf{c}_e^{(s)}\}_{e=1}^{K_s}$. After normalizing each center as $\bar{\mathbf{c}}_e^{(s)} = \mathbf{c}_e^{(s)}/\|\mathbf{c}_e^{(s)}\|_2$, we compute the affinity and identify the primary cluster as:

$$a_{ie}^{(s)} = \mathbf{y}_i^\top \bar{\mathbf{c}}_e^{(s)}, \quad e_i^{(s,\star)} = \arg\max_{1\le e\le K_s} a_{ie}^{(s)}. \tag{3}$$

We retain the top-κ most similar cluster centers $\mathcal{T}_i^{(s)}$ for node v_i and compute soft incidence weights:

$$H_{ie}^{(s)} = \exp\Big((a_{ie}^{(s)} - a_{i,\star}^{(s)})/\tau_h\Big), \quad \forall e \in \mathcal{T}_i^{(s)}, \tag{4}$$

where $H_{ie}^{(s)} = 0$ for $e \notin \mathcal{T}_i^{(s)}$, $a_{i,\star}^{(s)} = \max_{1 \le e \le K_s} a_{ie}^{(s)}$, and $\tau_h > 0$ is a temperature parameter controlling the concentration of incidence weights. Concatenating the incidence matrices across all scales $M = \sum_{s=1}^{S} K_s$ yields the complete semantic incidence matrix $\mathbf{H} = \left[\mathbf{H}^{(1)} \mid \cdots \mid \mathbf{H}^{(S)}\right]$, where $\mathbf{H}$ captures overlapping, group-level semantic relations at coarse-to-fine resolutions.

Membership Reliability. We estimate membership reliability from assignment separation and stability across T perturbed attribute views:

$$\Delta_i^{(s)} = \mathrm{clip}\Big((a_{i,(1)}^{(s)} - a_{i,(2)}^{(s)})/\tau_m, 0, 1\Big), \qquad q_i^{(s)} = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}\left[e_i^{(s,\star,t)} = e_i^{(s,\star)}\right]. \tag{5}$$

where $a_{i,(1)}^{(s)}$ and $a_{i,(2)}^{(s)}$ are the two largest affinities, and cluster centers are shared across views. The perturbation operator and the values of T, τ_m, and the SVD-related parameters are provided in Appendix A.2.1. We define c_i as Eqn. (6), which measures assignment consistency.

$$c_i = \mathrm{clip}\left(\frac{1}{2S}\sum_{s=1}^{S}(\Delta_i^{(s)} + q_i^{(s)}), c_{\min}, 1\right). \tag{6}$$

3.4 Hypergraph Teacher Pre-training and Contextual Guidance

Teacher Pre-training. We pretrain the hypergraph teacher to reconstruct masked node attributes on the semantic incidence matrix $\mathbf{H}$. The masked nodes are selected using a hybrid masking strategy that combines a hyperedge-correlated component and an independent node-level component.

For the hyperedge randomness, we sample a random variable $\xi_e \sim \mathcal{U}(0,1)$ for each semantic hyperedge and calculate the incidence-weighted average of the random variables associated with v_i:

$$g_i = \frac{\sum_{e=1}^{M} H_{ie}\xi_e}{\max\left(\sum_{e=1}^{M} H_{ie}, \epsilon\right)}, \tag{7}$$

where $\epsilon > 0$ ensures numerical stability. Nodes with similar hyperedge memberships share more of the same hyperedge-level random variables and therefore tend to receive correlated masking scores.

For the node randomness, each node is assigned a random score $u_i \sim \mathcal{U}(0,1)$. The two components are combined before node selection:

$$s_i^{\mathrm{mask}} = \rho_h g_i + (1-\rho_h)u_i, \qquad \rho_h \in [0,1], \tag{8}$$

where ρ_h controls the contribution of hyperedge-correlated randomness. Let $N_m = \max(1, \lfloor \rho_m N \rfloor)$, where ρ_m is the attribute mask rate. The N_m nodes with the largest s_i^{mask} values form the single mask set $\mathcal{V}_{\mathrm{mask}}^{H}$. Their attributes are replaced with a learnable mask token $\mathbf{x}_{\mathrm{mask}}$ to construct $\widetilde{\mathbf{X}}$. The hypergraph teacher then encodes the masked view and reconstructs the original attributes: $\mathbf{Z}_H = f_H(\widetilde{\mathbf{X}}, \mathbf{H}), \widehat{\mathbf{X}}_H = D_H(\mathbf{Z}_H)$. The teacher is optimized using the scaled cosine error over the masked nodes:

$$\mathcal{L}_{\mathrm{tea}} = \mathcal{L}_{\mathrm{sce}}\left(\widehat{\mathbf{X}}_H, \mathbf{X}; \mathcal{V}_{\mathrm{mask}}^{H}\right). \tag{9}$$

The trained teacher is frozen and placed in evaluation mode for student pre-training.

Contextual Hyperedge Prototypes. During student pre-training, the frozen teacher provides higher-order semantic guidance. Given the masked attribute view $\tilde{\mathbf{X}}$ shared with the student, the frozen teacher extracts contextual representations $Z_H^{\mathrm{ctx}} = f_H(\tilde{\mathbf{X}}, \mathbf{H})$. Let $\mathbf{H}_c = \mathrm{diag}(c)\mathbf{H}$ denote the reliability-weighted incidence matrix and $D_{e,c} = \mathrm{diag}(\mathbf{H}_c^{\top}\mathbf{1}_N)$ be its diagonal hyperedge degree matrix. We construct reliability-weighted contextual hyperedge prototypes as:

$$\mathbf{P}^{\mathrm{ctx}} = \mathbf{D}_{e,c}^{-1}\mathbf{H}_c^{\top}\mathbf{Z}_H^{\mathrm{ctx}}, \tag{10}$$

and subsequently project them back to obtain node-level semantic prompts:

$$\mathbf{Q} = \mathbf{D}_v^{-1}\mathbf{H}\mathbf{P}^{\mathrm{ctx}}, \qquad \mathbf{D}_v = \mathrm{diag}(\mathbf{H}\mathbf{1}_M), \tag{11}$$

where $\mathbf{q}_i$ summarizes the multi-scale semantic contexts for node v_i. For a fixed membership weight, a node with a higher c_i receives a larger relative contribution to its incident hyperedge prototype.

3.5 RELIABILITY-WEIGHTED GRAPH STUDENT

Observed-edge Reliability. For each observed edge $(i, j) \in \mathcal{E}_o$, we estimate its topological reliability by combining pairwise attribute agreement and higher-order semantic consistency. Let $\bar{x}_i = x_i/\|x_i\|_2$ denotes the normalized attribute vector and $\bar{h}_i = H_i/\|H_i\|_2$ denotes the hypergraph membership profile of node v_i. The attribute agreement and semantic consistency are defined as:

$$s_{ij}^x = \text{clip}\left(\overline{\mathbf{x}}_i^\top \overline{\mathbf{x}}_j, 0, 1\right), \quad s_{ij}^h = \text{clip}\left(\overline{\mathbf{h}}_i^\top \overline{\mathbf{h}}_j, 0, 1\right), \tag{12}$$

where s_{ij}^x measures local attribute agreement, whereas semantic hypergraph co-membership reflects multi-scale co-membership across the semantic hypergraph. Combining both sources of representations yields the edge reliability score r_{ij} and the resulting propagation weight w_{ij}:

$$r_{ij} = \left[\lambda_e s_{ij}^h + (1 - \lambda_e) s_{ij}^x\right]^\gamma, \qquad w_{ij} = \epsilon_e + (1 - \epsilon_e) r_{ij}, \tag{13}$$

where $\lambda_e \in [0, 1]$ balances the two criteria, γ controls the concentration of the reliability weights, and $\epsilon_e \in [0, 1]$ controls a residual floor on observed-edge weights.

We construct the observed weight matrix $\mathbf{W}_o = [w_{ij} \cdot \mathbb{I}((i, j) \in \mathcal{E}_o)]$ and denote its symmetrically normalized form with self-loops as $\widehat{\mathbf{W}}_o$. The graph student encoder f_G propagates the masked attribute view $\tilde{X}$ over the reweighted topology:

$$\mathbf{Z}_G = f_G(\widetilde{\mathbf{X}}, \widehat{\mathbf{W}}_o). \tag{14}$$

To integrate topology-independent guidance into graph learning, we first combine the frozen teacher's representation $z_{H,i}^{\text{ctx}}$ and the node-level contextual prompt q_i:

$$\mathbf{h}_i^{\text{sem}} = \lambda_h \mathbf{z}_{H,i}^{\text{ctx}} + (1 - \lambda_h)\mathbf{q}_i, \tag{15}$$

where $\lambda_h \in [0, 1]$ balances their contributions.

Representation Discrepancy. We use the normalized cosine discrepancy between the graph-student and frozen-teacher representations as an auxiliary input:

$$d_i = \frac{1}{2}\left(1 - \frac{\mathbf{z}_{G,i}^\top \mathbf{z}_{H,i}^{\text{ctx}}}{\|\mathbf{z}_{G,i}\|_2 \|\mathbf{z}_{H,i}^{\text{ctx}}\|_2}\right), \tag{16}$$

We combine it with the attribute-derived reliability score as $\eta_i = (c_i + d_i)/2$.

3.6 NODE-AWARE ADAPTIVE REPRESENTATION FUSION

We introduce a node-aware adaptive fusion strategy to control the contribution of semantic guidance during student pre-training. Its input combines the two representations and their element-wise absolute difference:

$$\mathbf{u}_i = \left[\mathbf{z}_{G,i} \parallel \mathbf{z}_{H,i}^{\text{ctx}} \parallel \left|\mathbf{z}_{G,i} - \mathbf{z}_{H,i}^{\text{ctx}}\right|\right]. \tag{17}$$

The graph student and fused decoder are jointly optimized against the fixed teacher guidance. The semantic fusion weight is calculated as follows:

$$\beta_i = s_f\, \sigma\left(g_\omega(\text{sg}(\mathbf{u}_i)) + 2\left(\text{sg}(\eta_i) - \frac{1}{2}\right)\right), \quad s_f \in [0, 1], \tag{18}$$

where $g_\omega(\cdot)$ is a two-layer MLP, $\sigma(\cdot)$ is the sigmoid function, and $\text{sg}(\cdot)$ denotes stop-gradient. The scale s_f bounds the contribution of semantic guidance.

The fused representation is $\mathbf{z}_{F,i} = \text{LN}[(1 - \beta_i)\mathbf{z}_{G,i} + \beta_i \mathbf{h}_i^{\text{sem}}]$, where LN denotes layer normalization. This combination is trained through the fused reconstruction objective.

3.7 OPTIMIZATION AND INFERENCE

Optimization Loss. We apply attribute decoders D_F and D_G to the fused representation $\mathbf{Z}_F$ and the graph-student representation $\mathbf{Z}_G$, respectively. Both branches reconstruct the original attributes of the same masked node set $\mathcal{V}_{\text{mask}}$:

$$\begin{aligned} \mathcal{L}_{\text{fused}} &= \mathcal{L}_{\text{sce}}\left(D_F(\mathbf{Z}_F), \mathbf{X}; \mathcal{V}_{\text{mask}}\right), \\ \mathcal{L}_{\text{graph}} &= \mathcal{L}_{\text{sce}}\left(D_G(\mathbf{Z}_G), \mathbf{X}; \mathcal{V}_{\text{mask}}\right). \end{aligned} \tag{19}$$

The fused representations train the student together with the fusion module using semantic guidance. The graph-specific representation provides direct reconstruction supervision to the student without passing through the fusion module. The overall student optimization loss is shown in Eqn. (20).

$$\mathcal{L}_{\text{stu}} = \mathcal{L}_{\text{fused}} + \lambda_{\text{recon}}\mathcal{L}_{\text{graph}}, \tag{20}$$

where λ_{recon} controls the strength of graph-specific reconstruction. We update the graph student, fusion normalization, reconstruction decoders, and learnable mask token.

Downstream Inference. After pre-training, we extract node representations from the graph student encoder for downstream tasks.

$$\mathbf{Z}_{\text{eval}} = \text{RowNorm}\left(f_G(\mathbf{X}, \widehat{\mathbf{W}}_o(\mathbf{A}_o, \mathbf{X}, \mathbf{H}))\right). \tag{21}$$

where RowNorm denotes row-wise ℓ_2 normalization and $\mathbf{W}_o$ is constructed using Eqn. (13). A linear classifier is trained on the resulting frozen embeddings using the training labels.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate the proposed method on five datasets, including three citation networks (Cora, Citeseer, and PubMed) (Yang et al., 2016) and two Amazon co-purchase networks (Photo and Computers) (Shchur et al., 2018). Additional experiments on two industrial control-system datasets are provided in Appendix A.4. The statistical results of the datasets are shown in Table 4.

Baselines & Implementations. We compare HyperRoGMAE with several representative graph masked autoencoders on the task of node classification against structural attacks, including GraphMAE (Hou et al., 2022), MaskGAE (Li et al., 2023), Bandana (Zhao et al., 2024), AUGMAE (Wang et al., 2024a), GCMAE (Wang et al., 2024b), ProtoMGAE (Zheng & Jia, 2024), AMGAE (Hu et al., 2025), DGMAE (Zheng et al., 2025), and ArmorGAE (Zhang et al., 2026). Meanwhile, we evaluate HyperRoGMAE for anomaly detection using LSTM-AE (Malhotra et al., 2016) and TCN-AE (Thill et al., 2021). Descriptions of the baselines and their implementation details are provided in Appendix A.4.

Attack Settings. We evaluate the robustness of the proposed method under four structural attacks: Metattack (Zügner & Günnemann, 2019), Nettack (Zügner et al., 2018), random edge insertion, and DICE (Waniek et al., 2018). We consider three evaluation scenarios: *clean*, where both pre-training and evaluation use the clean graph; *poisoning*, where both pre-training and evaluation use the attacked graph; and *evasion*, where the model is pre-trained on the clean graph but evaluated on an attacked topology. Results for all evaluated ratios are provided in Appendix A.5.

4.2 Experimental Results

Robustness against Metattack. As shown in Table 1, HyperRoGMAE exhibits superior robustness against non-targeted attacks, achieving the best results in 23 out of 35 settings and recording the highest average accuracy on Cora, Citeseer, PubMed, and Photo datasets. Specifically, it outperforms the strongest baseline on these four datasets by 7.99%, 2.94%, 5.25%, and 2.15%, respectively. On Computers, HyperRoGMAE remains highly competitive, ranking second overall and achieving the top accuracy across all evasion settings. This superiority demonstrates that the topology-independent semantic hypergraph and adaptive edge fusion strategy effectively mitigate structural noise from degrading the learned representations.

Robustness against Nettack. As reported in Table 2, HyperRoGMAE delivers superior defense against targeted structural perturbations, attaining the top performance across the majority of attacked scenarios and recording the highest average accuracy on Cora, Citeseer, and PubMed with gains of 2.55%, 5.85%, and 2.32% over the SOTA baselines. At a perturbation ratio of 5, it surpasses the second-best method by 6.57%, 19.52%, and 5.93% under poisoning attacks and by 3.38%, 4.77%, and 2.94% under evasion attacks. This superiority demonstrates that decoupling high-order semantic references from the observed topology effectively mitigates targeted local adversarial connections from corrupting the target nodes' representations.

Table 1: Node classification accuracy (%) of HyperRoGMAE and five representative baselines under non-targeted attack (Metattack) in the clean, poisoning, and evasion settings.

Dataset	Method	Clean	Poisoning			Evasion			Avg.
			5%	15%	25%	5%	15%	25%	
Cora	AUGMAE'24	83.25±0.00	77.73±0.35	65.10±0.31	48.80±0.38	80.35±1.00	73.27±0.28	60.76±0.57	69.89
	ProtoMGAE'24	83.56±0.35	77.86±0.16	62.97±0.56	47.39±1.04	78.80±0.55	61.30±0.98	44.95±0.57	65.26
	AMGAE'25	83.52±0.45	77.64±0.81	65.02±0.64	47.49±0.82	81.08±0.44	74.09±0.77	58.60±0.97	69.63
	DGMAE'25	83.45±0.08	78.48±0.16	68.31±0.15	50.68±0.19	81.05±0.11	74.12±0.17	58.47±0.12	70.65
	ArmorGAE'26	**85.53±0.32**	**81.25±0.33**	71.37±0.45	52.28±1.37	**81.87±0.23**	75.16±1.35	57.79±0.44	72.18
	HyperRoGMAE	83.48±0.46	80.46±0.11	**78.85±0.31**	**78.59±0.72**	80.06±0.47	**79.95±0.13**	**79.77±0.09**	**80.17**
Citeseer	AUGMAE'24	73.25±0.22	71.47±0.20	66.18±0.36	56.62±0.33	72.59±0.16	71.02±0.29	66.72±0.27	68.26
	ProtoMGAE'24	74.28±0.31	73.36±0.36	68.60±0.55	60.34±0.79	72.53±0.69	68.00±1.61	62.84±1.76	68.56
	AMGAE'25	73.86±0.27	71.71±0.33	65.65±0.76	56.21±0.91	73.16±0.28	70.95±0.84	66.86±0.55	68.34
	DGMAE'25	74.82±0.00	72.93±0.00	69.37±0.00	59.60±0.00	73.82±0.00	70.26±0.00	65.88±0.00	69.53
	ArmorGAE'26	**76.05±0.15**	**75.59±0.14**	71.94±0.32	64.14±1.65	**74.97±0.10**	72.32±0.36	66.18±0.58	71.60
	HyperRoGMAE	75.50±0.12	74.96±0.27	**74.19±0.20**	**72.76±0.53**	74.89±0.11	**74.85±0.22**	**74.66±0.15**	**74.54**
PubMed	AUGMAE'24	82.47±0.02	80.59±0.27	77.81±0.35	74.07±0.62	81.00±0.02	76.30±0.03	71.13±0.04	77.62
	ProtoMGAE'24	82.87±0.50	82.00±0.13	80.02±0.17	76.23±0.44	81.39±0.18	77.87±0.37	73.33±0.40	79.10
	AMGAE'25	83.04±0.10	81.67±0.18	78.53±0.31	75.06±0.39	81.82±0.08	78.07±0.17	73.70±0.13	78.84
	DGMAE'25	83.51±0.01	82.68±0.03	81.55±0.05	77.85±0.04	79.83±0.12	77.70±0.20	71.30±0.03	79.20
	ArmorGAE'26	**85.19±0.06**	82.14±0.06	78.73±0.21	76.08±0.15	81.73±0.09	76.93±0.14	72.27±0.14	79.01
	HyperRoGMAE	84.69±0.03	**84.57±0.06**	**84.27±0.16**	**84.09±0.04**	**84.61±0.05**	**84.55±0.04**	**84.34±0.06**	**84.45**
Photo	AUGMAE'24	93.10±0.36	90.49±0.28	89.02±0.24	88.15±0.51	90.63±0.35	89.72±0.40	88.85±0.49	89.99
	ProtoMGAE'24	92.34±0.26	90.26±0.15	89.70±0.48	89.63±0.31	90.46±0.21	90.24±0.50	89.94±0.36	90.37
	DGMAE'25	92.58±0.66	90.47±0.60	89.35±0.88	88.49±1.38	90.23±0.50	89.23±0.56	88.61±0.51	89.85
	AMGAE'25	91.97±0.59	87.93±1.54	86.60±1.39	87.07±2.39	88.29±3.96	87.39±4.30	87.02±3.50	88.04
	ArmorGAE'26	**93.46±0.26**	89.77±0.60	89.33±0.32	89.11±0.26	90.34±0.31	89.69±0.24	88.93±0.27	90.09
	HyperRoGMAE	92.98±0.06	**92.33±0.09**	**92.29±0.07**	**92.24±0.09**	**92.78±0.14**	**92.61±0.03**	**92.42±0.06**	**92.52**
Computers	AUGMAE'24	87.70±0.59	85.37±0.26	77.93±0.47	72.44±0.92	84.95±0.37	83.62±0.28	82.54±0.41	82.08
	ProtoMGAE'24	**89.57±0.31**	87.49±0.34	86.32±0.22	85.70±0.16	87.28±0.30	86.24±0.32	85.53±0.24	86.88
	DGMAE'25	87.50±0.41	84.62±1.72	82.98±1.56	80.79±2.75	85.08±0.55	83.37±0.42	82.23±0.55	83.80
	AMGAE'25	88.99±0.23	86.78±0.58	84.87±0.11	84.22±0.40	86.87±0.22	85.04±0.27	84.17±0.43	85.85
	ArmorGAE'26	89.53±0.26	**89.35±0.13**	**88.38±0.27**	**88.25±0.16**	86.66±0.31	85.49±0.28	85.09±0.30	**87.54**
	HyperRoGMAE	87.55±0.04	87.27±0.12	86.76±0.04	86.62±0.08	**87.36±0.03**	**87.01±0.06**	**86.50±0.06**	87.01

Table 2: Node classification accuracy (%) of HyperRoGMAE and five representative baselines under targeted attack (Nettack) in the clean, poisoning, and evasion settings.

Dataset	Method	Clean	Poisoning			Evasion			Avg.
			1	3	5	1	3	5	
Cora	AUGMAE'24	82.75±1.10	79.36±1.30	71.69±0.78	63.98±0.98	81.45±0.59	76.11±0.84	69.88±0.93	75.03
	ProtoMGAE'24	83.42±0.66	79.66±0.80	72.26±1.47	63.39±1.34	79.06±1.08	74.96±0.92	70.90±0.94	74.81
	AMGAE'25	82.17±1.57	78.80±1.62	74.94±1.32	63.37±1.08	80.24±1.83	**76.87±2.15**	71.08±1.91	75.35
	DGMAE'25	83.78±1.13	76.89±1.02	72.26±0.62	66.92±0.74	80.12±0.97	73.25±0.73	68.07±0.63	74.47
	ArmorGAE'26	**85.30±1.55**	80.12±1.08	73.37±0.59	62.17±1.80	**83.98±0.18**	75.18±1.37	66.63±1.21	75.25
	HyperRoGMAE	82.17±1.93	**81.20±0.59**	**75.90±2.02**	**73.49±0.00**	82.17±0.48	75.90±1.32	**74.46±1.18**	**77.90**
Citeseer	AUGMAE'24	80.95±0.00	80.95±0.00	65.34±1.62	60.48±3.97	79.37±0.00	65.24±0.50	63.49±0.00	70.83
	ProtoMGAE'24	71.73±0.46	81.27±1.33	59.37±2.40	49.95±2.51	73.59±5.04	56.55±3.35	40.59±4.71	61.86
	AMGAE'25	79.69±0.71	79.69±0.71	65.08±4.05	56.51±1.42	79.37±0.00	64.13±0.87	63.49±0.00	69.71
	DGMAE'25	**82.54±0.00**	80.95±0.00	61.90±0.00	55.56±0.00	80.95±0.00	66.67±0.00	58.73±0.00	69.61
	ArmorGAE'26	**82.54±0.21**	**81.79±0.48**	64.92±0.27	58.25±2.66	**82.86±0.00**	79.37±0.73	74.60±2.13	74.90
	HyperRoGMAE	**82.54±0.00**	80.95±0.00	**80.78±0.00**	**80.00±0.78**	80.95±0.00	**80.63±0.63**	**79.37±0.00**	**80.75**
PubMed	AUGMAE'24	87.79±0.51	85.22±0.26	83.33±0.12	78.37±0.23	87.10±0.93	84.41±0.81	80.07±0.51	83.76
	ProtoMGAE'24	80.19±0.40	87.19±1.36	80.85±0.80	75.64±1.58	87.26±0.93	84.19±0.98	80.07±0.51	82.20
	AMGAE'25	**88.35±0.70**	**87.39±0.98**	82.19±0.48	77.74±0.72	87.49±0.62	84.38±0.59	80.72±0.85	84.04
	DGMAE'25	87.25±0.54	86.48±0.56	81.72±0.98	76.96±1.98	87.17±0.80	83.87±0.61	76.56±1.17	82.86
	ArmorGAE'26	88.15±0.47	87.01±0.54	80.32±0.92	77.92±0.35	87.31±0.02	83.35±0.11	80.46±0.85	83.50
	HyperRoGMAE	87.63±0.00	87.31±0.26	**86.77±0.26**	**84.30±0.53**	**87.63±0.00**	**87.20±0.22**	**83.66±0.65**	**86.36**

Robustness against Random and DICE attacks. Figure 3 depicts the classification accuracy of evaluated methods across five different perturbation ratios under random and DICE attacks. HyperRoGMAE is slightly inferior to the strongest baseline on Cora at low poisoning ratios. Its advantage becomes increasingly evident as the ratio increases. At a 50% poisoning ratio, it outperforms the best baseline by 3.80% and 10.01% under DICE attacks on Cora and Citeseer, respectively, and by 3.83% and 10.67% under random attacks. From the clean setting to 50% poisoning, its accuracy decreases by only 10.51%, 3.42%, 4.17%, and 0.61% across the four settings, substantially less than the competing methods. These results demonstrate that integrating edge-reliability filtering prevents structural noise from contaminating the representations.

Ablation Study. We conduct ablation studies on Cora and Citeseer under Metattack with a 25% perturbation rate. There are three variants: edge reliability weighting (*ER*), graph-specific attribute reconstruction (*GR*), and semantic teacher fusion (*TF*). As shown in Figure 4(a), removing *ER* causes the largest accuracy reductions. This result identifies reliability-weighted message passing as play-

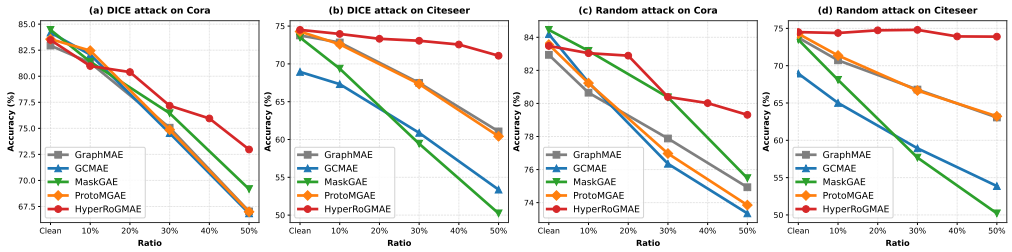
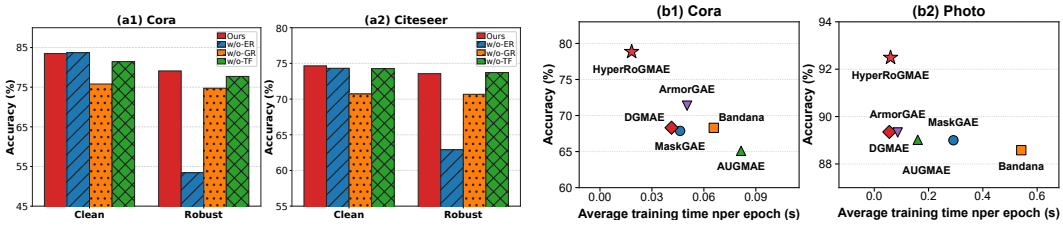


Figure 3: Node classification accuracy (%) of HyperRoGMAE under Random and DICE attacks.



(a) Results for the ablation study.

(b) Efficiency comparison of different methods.

Figure 4: Ablation study and performance-efficiency comparison of HyperRoGMAE.

ing an important role in improving robustness. Removing *GR* also consistently reduces accuracy, showing that the auxiliary graph-specific reconstruction helps the student retain information. The ablation of *TF* further highlights the complementary role of semantic teacher fusion.

Training Efficiency. Figure 4(b) compares the accuracy and average training time per epoch of six methods on Cora and Photo under MetaAttack with a 15% perturbation rate in the poisoning setting. HyperRoGMAE achieves the highest accuracy among the compared methods on both datasets. On Cora, it improves upon the strongest baseline, ArmorGAE, by 7.48% while requiring an average of 18.30 ms per epoch, compared ArmorGAE (50.30 ms). On Photo, HyperRoGMAE outperforms the strongest baseline, DGMAE, by 3.14%, with a modest increase in average epoch time from 55.52 ms to 60.00 ms. These results indicate a favorable balance between robustness and training efficiency.

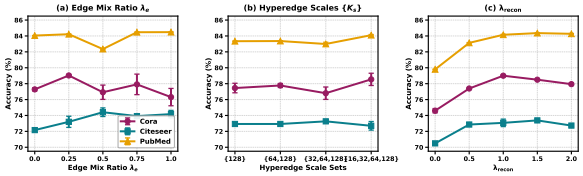


Figure 5: Hyper-parameter analysis.

Parameter Analysis. *(i) Edge Mix Ratio (λ_e)*: Figure 5(a) shows that the best performance is achieved at $\lambda_e = 0.25$ on Cora, 0.5 on Citeseer, and 0.75 on PubMed, suggesting the preferred balance between feature similarity and hypergraph-membership consistency. *(ii) Hyperedge Scales ($\{K_s\}$)*: Figure 5(b) shows relatively stable performance across scale configurations, with $\{16, 32, 64, 128\}$ yielding the highest accuracy on Cora and PubMed, whereas Citeseer favors $\{32, 64, 128\}$. *(iii) Weight (λ_{recon})*: Figure 5(c) shows that accuracy improves compared with $\lambda_{\text{recon}} = 0$ and peaks in the range of 1.0-1.5 across these datasets, supporting the use of a moderate weight for graph-specific reconstruction.

5 Conclusion

In this paper, we propose a reliability-aware hypergraph-guided framework for robust graph masked autoencoder, named HyperRoGMAE. We construct an attribute-induced multi-scale semantic hypergraph as a topology-independent reference and utilize a frozen hypergraph teacher to provide higher-order contextual guidance. Moreover, we suppress suspicious observed edges via feature- and membership-consistency-based reliability weighting. Finally, we propose a node-aware adaptive fusion strategy to regulate the influence of semantic guidance during pre-training. Experimental results demonstrate the robustness of HyperRoGMAE under four structural attacks.

AI USE STATEMENT

During the preparation of this manuscript, the authors used a large language model for language polishing and grammatical refinement of author-drafted text. The AI tool was not used to generate core research ideas, model designs, experimental schemes or conclusions. All AI-suggested texts were fully reviewed, revised and verified manually by the authors. The authors take full responsibility for all content in this paper.

REFERENCES

Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are AllSet: A multiset function framework for hypergraph neural networks. In *International Conference on Learning Representations*, 2022.

Shengyu Feng, Baoyu Jing, Yada Zhu, and Hanghang Tong. ArieL: Adversarial graph contrastive learning. *ACM Transactions on Knowledge Discovery from Data*, 18(4):1–22, 2024.

Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(1):3558–3565, 2019.

Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. GraphMAE: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 594–604, 2022.

Zhenyu Hou, Yufei He, Yukuo Cen, Xiao Liu, Yuxiao Dong, Evgeny Kharlamov, and Jie Tang. GraphMAE2: A decoding-enhanced masked self-supervised graph learner. In *Proceedings of the ACM Web Conference 2023*, pp. 737–746, 2023.

Yulan Hu, Zhirui Yang, Sheng Ouyang, and Yong Liu. Adversarial masked graph autoencoders for improved graph representation learning. In *Proceedings of the 2025 International Conference on Multimedia Retrieval*, pp. 478–486, 2025.

Yiming Huang and Tolga Birdal. HOG-Diff: Higher-order guided diffusion for graph generation. In *International Conference on Learning Representations*, 2026.

Yeonjun In, Kanghoon Yoon, and Chanyoung Park. Similarity preserving adversarial graph contrastive learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 867–878, 2023.

Wei Jin, Yao Ma, Xiaorui Liu, Xianfeng Tang, Suhang Wang, and Jiliang Tang. Graph structure learning for robust graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 66–74, 2020.

Sunwoo Kim, Shinhwan Kang, Fanchen Bu, Soo Yong Lee, Jaemin Yoo, and Kijung Shin. HypeBoy: Generative self-supervised representation learning on hypergraphs. In *International Conference on Learning Representations*, 2024.

Dongjin Lee and Kijung Shin. I'm me, we're us, and i'm us: Tri-directional contrastive learning on hypergraphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(7):8456–8464, 2023.

Jintang Li, Ruofan Wu, Wangbin Sun, Liang Chen, Sheng Tian, Liang Zhu, Changhua Meng, Zibin Zheng, and Weiqiang Wang. What's behind the mask: Understanding masked graph modeling for graph autoencoders. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1268–1279, 2023.

Kuan Li, Yang Liu, Xiang Ao, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Reliable representations make a stronger defender: Unsupervised structure refinement for robust GNN. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 925–935, 2022.

Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, and Gautam Shroff. LSTM-based encoder-decoder for multi-sensor anomaly detection. In *Workshop on Anomaly Detection at the International Conference on Machine Learning*, 2016.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. In *Relational Representation Learning Workshop at the Conference on Neural Information Processing Systems*, 2018.

Markus Thill, Wolfgang Konen, Hao Wang, and Thomas Bäck. Temporal convolutional autoencoder for unsupervised anomaly detection in time series. *Applied Soft Computing*, 112:107751, 2021.

Liang Wang, Xiang Tao, Qiang Liu, Shu Wu, and Liang Wang. Rethinking graph masked autoencoders through alignment and uniformity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 15528–15536, 2024a.

Yuxiang Wang, Xiao Yan, Chuang Hu, Quanqing Xu, Chuanhui Yang, Fangcheng Fu, Wentao Zhang, Hao Wang, Bo Du, and Jiawei Jiang. Generative and contrastive paradigms are complementary for graph self-supervised learning. In *Proceedings of the 40th IEEE International Conference on Data Engineering*, pp. 3364–3378, 2024b.

Marcin Waniek, Tomasz P. Michalak, Michael J. Wooldridge, and Talal Rahwan. Hiding individuals and communities in a social network. *Nature Human Behaviour*, 2:139–147, 2018.

Tianxin Wei, Yuning You, Tianlong Chen, Yang Shen, Jingrui He, and Zhangyang Wang. Augmentations in hypergraph contrastive learning: Fabricated and generative. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1909–1922, 2022.

Huijun Wu, Chen Wang, Yuriy Tyshetskiy, Andrew Docherty, Kai Lu, and Liming Zhu. Adversarial examples for graph data: Deep insights into attack and defense. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4816–4823, 2019.

Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha P. Talukdar. HyperGCN: A new method for training graph convolutional networks on hypergraphs. In *Advances in Neural Information Processing Systems*, volume 32, pp. 1509–1520, 2019.

Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. Revisiting semi-supervised learning with graph embeddings. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 40–48. PMLR, 2016.

Qiqi Zhang, Chuanjin Liu, Gen Liu, Chao Li, and Zhongying Zhao. Adversarial masked graph modeling for robust graph autoencoders. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 5315–5323, 2026. doi: 10.24963/ijcai.2026/592.

Xiang Zhang and Marinka Zitnik. GNNGuard: Defending graph neural networks against adversarial attacks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9263–9275, 2020.

Zhongying Zhao, Zhan Yang, Chao Li, Qingtian Zeng, Weili Guan, and MengChu Zhou. Dual feature interaction-based graph convolutional network. *IEEE Transactions on Knowledge and Data Engineering*, 35(9):9019–9030, 2023. doi: 10.1109/TKDE.2022.3220789.

Ziwen Zhao, Yuhua Li, Yixiong Zou, Jiliang Tang, and Ruixuan Li. Masked graph autoencoder with non-discrete bandwidths. In *Proceedings of the ACM Web Conference 2024*, pp. 377–388, 2024.

Yimei Zheng and Caiyan Jia. ProtoMGAE: Prototype-aware masked graph auto-encoder for graph representation learning. *ACM Transactions on Knowledge Discovery from Data*, 18(6):1–22, 2024.

Ziyu Zheng, Yaming Yang, Ziyu Guan, Wei Zhao, and Weigang Lu. Discrepancy-aware graph mask auto-encoder. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 4038–4049, 2025.

Dingyuan Zhu, Ziwei Zhang, Peng Cui, and Wenwu Zhu. Robust graph convolutional networks against adversarial attacks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1399–1407, 2019.

Daniel Zügner and Stephan Günnemann. Adversarial attacks on graph neural networks via meta learning. In *International Conference on Learning Representations*, 2019.

Daniel Zügner, Amir Akbarnejad, and Stephan Günnemann. Adversarial attacks on neural networks for graph data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2847–2856, 2018.

A APPENDIX

A.1 NOTATIONS

The notations mainly used in this paper are summarized in Table 3.

Table 3: The notations used in this paper.

Notations	Descriptions
$\mathcal{G}_o$	The observed attributed graph.
$\mathcal{V}$	The set of nodes.
$\mathcal{E}_o$	The set of observed pairwise edges.
$\mathbf{A}_o$	The adjacency matrix of $\mathcal{G}_o$.
$\mathbf{X} \in \mathbb{R}^{N\times D}$	The node attribute matrix of $\mathcal{G}_o$, where D is the attribute dimension.
$\mathbf{x}_i \in \mathbb{R}^D$	The original attribute vector of node v_i.
$\mathcal{V}_{\text{mask}} \subseteq \mathcal{V}$	The masked node set selected during student pre-training.
$\mathcal{V}^H_{\text{mask}} \subseteq \mathcal{V}$	The masked node set selected during teacher pre-training.
$\widehat{\mathbf{X}} \in \mathbb{R}^{N\times D}$	The masked attribute matrix.
$\mathbf{x}_{\text{mask}} \in \mathbb{R}^D$	The learnable token used to replace the attributes of masked nodes.
$\mathcal{G}_s = (\mathcal{V}, \mathcal{E}_s)$	The attribute-induced semantic hypergraph.
$\mathcal{E}_s$	The set of semantic hyperedges.
$\mathbf{H} \in \mathbb{R}^{N\times M}$	The weighted node-hyperedge incidence matrix.
S	The number of semantic clustering scales.
K_s	The number of semantic clusters at scale s, with $M = \sum_{s=1}^{S} K_s$.
κ	The number of retained hyperedge memberships per node at each scale.
$\mathbf{D}_v \in \mathbb{R}^{N\times N}$	The diagonal weighted node-degree matrix.
$\mathbf{D}_e \in \mathbb{R}^{M\times M}$	The diagonal weighted hyperedge-degree matrix.
$c_i \in [c_{\min}, 1]$	The semantic reliability score of node v_i.
d	The node representation dimension.
f_H	The hypergraph teacher encoder.
D_H	The attribute decoder used to pretrain the hypergraph teacher.
f_G	The reliability-weighted graph student encoder.
D_G, D_F	The graph-specific and fused attribute decoders, respectively.
$\mathbf{Z}_H^{\text{ctx}} \in \mathbb{R}^{N\times d}$	The contextual representations.
$\mathbf{Z}_G \in \mathbb{R}^{N\times d}$	The graph-student representations.
$\mathbf{P}^{\text{ctx}} \in \mathbb{R}^{M\times d}$	The reliability-weighted contextual hyperedge prototypes.
$\mathbf{Q} \in \mathbb{R}^{N\times d}$	The node-level semantic prompts.
$s^x_{ij} \in [0, 1]$	The attribute agreement associated with observed edge (i, j).
$s^h_{ij} \in [0, 1]$	The hypergraph-membership consistency associated with observed edge (i, j).
$r_{ij} \in [0, 1]$	The observed-edge reliability score.
$w_{ij} \in [\epsilon_e, 1]$	The propagation weight assigned to observed edge.
$\mathbf{W}_o \in \mathbb{R}^{N\times N}$	The weighted observed adjacency matrix.
$\mathbf{h}_i^{\text{sem}} \in \mathbb{R}^d$	The semantic guidance vector.
$\beta_i \in [0, s_f]$	The node-aware semantic fusion weight.
$\mathbf{Z}_F \in \mathbb{R}^{N\times d}$	The fused node representations.
$\mathcal{L}_{\text{tea}}$	The teacher reconstruction loss during pre-training evaluated on $\mathcal{V}^H_{\text{mask}}$.
$\mathcal{L}_{\text{fused}}$	The masked attribute reconstruction loss computed using $D_F(\mathbf{Z}_F)$.
$\mathcal{L}_{\text{graph}}$	The masked attribute reconstruction loss computed using $D_G(\mathbf{Z}_G)$.
$\mathcal{L}_{\text{stu}}$	The overall student optimization loss.

A.2 Additional Methodological Details

This section provides additional implementation details for HyperRoGMAE introduced in Section 3.

A.2.1 Semantic Preprocessing and Hypergraph Construction

Attribute Projection. To obtain compact semantic representations before clustering, we apply truncated singular value decomposition (SVD) when dimensionality reduction is applicable. Let $d_s = 128$ denote the target projection dimension and define $r = \min(N-1, D-1, d_s)$ for the attribute matrix $\mathbf{X} \in \mathbb{R}^{N\times D}$. When $r \geq 2$ and $D > r$, we fit truncated SVD on $\mathbf{X}$ and retain the right singular vectors in $\mathbf{P}_{\mathrm{svd}} \in \mathbb{R}^{D\times r}$; otherwise, we retain the original feature space:

$$\mathbf{B} = \begin{cases} \mathbf{X}\mathbf{P}_{\mathrm{svd}}, & r \geq 2 \text{ and } D > r, \\ \mathbf{X}, & \text{otherwise.} \end{cases} \tag{22}$$

The resulting dimension is $d' = r$ when SVD is applied and $d' = D$ otherwise. The semantic representations used in Section 3.3 are obtained by row-wise ℓ_2 normalization:

$$\mathbf{y}_i = \begin{cases} \mathbf{b}_i / \|\mathbf{b}_i\|_2, & \|\mathbf{b}_i\|_2 > 0, \\ \mathbf{0}, & \text{otherwise,} \end{cases} \qquad \mathbf{Y} \in \mathbb{R}^{N\times d'}. \tag{23}$$

When SVD is applied, the fitted projection is held fixed throughout semantic reliability estimation, and no additional singular-value threshold is used.

Perturbed Attribute Views and Reliability Settings. At each clustering scale, we generate $T = 2$ independently perturbed attribute views to estimate cross-view assignment stability. Omitting the scale index for simplicity, we define:

$$\begin{aligned} M_{ij}^{(t)} &\sim \mathrm{Bernoulli}(1-\epsilon_x), \\ \mathbf{X}^{(t)} &= \mathbf{X} \odot \mathbf{M}^{(t)}, \qquad t = 1, \ldots, T, \end{aligned} \tag{24}$$

where $\mathbf{M}^{(t)} \in \{0,1\}^{N\times D}$, mask entries are sampled independently, $\odot$ denotes element-wise multiplication, and $\epsilon_x = 0.1$ is the feature-drop probability. Each perturbed view is transformed using the fixed SVD projection, when applicable, and normalized row-wise as described above. Its primary cluster assignments are computed using the fixed normalized cluster centers fitted on the original semantic representations at the corresponding scale, yielding $e_i^{(s,\star,t)}$ in Eqn. (5). We set the margin clipping scale to $\tau_m = 0.25$, the soft-incidence temperature to $\tau_h = 0.2$, and the semantic reliability lower bound to $c_{\min} = 0.05$.

Sparse Incidence and Normalization. Following Eqn. (4), each node retains at most κ nonzero memberships at each of the S scales. Thus, the multi-scale incidence matrix $\mathbf{H} \in \mathbb{R}^{N\times M}$ satisfies:

$$\mathrm{nnz}(\mathbf{H}) \leq N\kappa S, \qquad M = \sum_{s=1}^{S} K_s. \tag{25}$$

We use the degree matrices $\mathbf{D}_v$ and $\mathbf{D}_e$ defined in the main text for semantic hypergraph propagation, and $\mathbf{D}_{e,c}$ for reliability-weighted contextual prototype construction. In implementation, inverse degrees used in hypergraph propagation and contextual prompting are computed as $1/\max(\delta, \epsilon_{\deg})$, where δ denotes a diagonal degree entry and $\epsilon_{\deg} = 10^{-8}$.

A.2.2 Encoder Architectures and Layer Specifications

Hypergraph Teacher Encoder (f_H). The teacher performs message passing over the attribute-induced incidence matrix $\mathbf{H}$. Let $\mathbf{Z}_H^{(0)} = \widetilde{\mathbf{X}}$. At each layer ℓ, dropout is applied once, and the resulting representations are shared by the propagation and residual branches:

$$\overline{\mathbf{Z}}_H^{(\ell)} = \mathrm{Dropout}\left(\mathbf{Z}_H^{(\ell)}\right). \tag{26}$$

The layer first aggregates node representations into hyperedges and then propagates the resulting hyperedge representations back to nodes:

$$\begin{aligned} \mathbf{E}_H^{(\ell)} &= \mathbf{D}_e^{-1}\mathbf{H}^\top \overline{\mathbf{Z}}_H^{(\ell)}, \\ \mathbf{V}_H^{(\ell)} &= \mathbf{D}_v^{-1}\mathbf{H}\mathbf{E}_H^{(\ell)}. \end{aligned} \tag{27}$$

The layer output is:

$$\mathbf{Z}_H^{(\ell+1)} = \mathrm{ELU}\Big(\mathrm{LN}\Big(\mathbf{V}_H^{(\ell)}\mathbf{W}_H^{(\ell)} + \mathcal{R}^{(\ell)}\Big(\overline{\mathbf{Z}}_H^{(\ell)}\Big)\Big)\Big), \tag{28}$$

where $\mathbf{W}_H^{(\ell)}$ is a trainable weight matrix, LN denotes layer normalization, and $\mathcal{R}^{(\ell)}$ is a learnable linear projection when the input and output dimensions differ and the identity mapping otherwise. Both the propagation projection and the residual projection are implemented without bias.

After teacher pre-training using Eqn. (9), the selected teacher checkpoint is restored, and the teacher parameters are frozen. The teacher is placed in evaluation mode, disabling dropout during student pre-training.

Reliability-Weighted Graph Student (f_G). The student encoder propagates masked attributes over the reliability-reweighted observed topology. Following Eqn. (13), the weighted adjacency matrix is defined as:

$$(\mathbf{W}_o)_{ij} = \begin{cases} w_{ij}, & (i,j) \in \mathcal{E}_o, \\ 0, & \text{otherwise.} \end{cases} \tag{29}$$

We add self-loops and apply symmetric normalization:

$$\begin{aligned} \widetilde{\mathbf{W}}_o &= \mathbf{W}_o + \mathbf{I}, \\ \widetilde{\mathbf{D}} &= \mathrm{diag}\left(\widetilde{\mathbf{W}}_o \mathbf{1}_N\right), \\ \widehat{\mathbf{W}}_o &= \widetilde{\mathbf{D}}^{-1/2}\widetilde{\mathbf{W}}_o\widetilde{\mathbf{D}}^{-1/2}. \end{aligned} \tag{30}$$

Starting from $\mathbf{Z}_G^{(0)} = \widetilde{\mathbf{X}}$, each student layer updates the representations as:

$$\begin{aligned} \mathbf{V}_G^{(\ell)} &= \widehat{\mathbf{W}}_o\,\mathrm{Dropout}\Big(\mathbf{Z}_G^{(\ell)}\Big)\,\mathbf{W}_G^{(\ell)} + \mathbf{1}_N(\mathbf{b}_G^{(\ell)})^\top, \\ \mathbf{Z}_G^{(\ell+1)} &= \mathrm{ELU}\Big(\mathrm{LN}\Big(\mathbf{V}_G^{(\ell)}\Big)\Big), \end{aligned} \tag{31}$$

where $\mathbf{W}_G^{(\ell)}$ and $\mathbf{b}_G^{(\ell)}$ are a trainable weight matrix and bias vector, respectively. The teacher and student encoders use the same output dimension d, allowing their representations and the node-level semantic prompts to be combined.

Attribute Decoders (D_H, D_F, D_G). Each attribute decoder maps d-dimensional latent representations to the original D-dimensional attribute space. The decoder D_H is used for teacher pre-training, whereas D_F and D_G reconstruct attributes from the fused and graph-student representations, respectively.

A.2.3 Optimization and Gradient Safeguards

Dynamic Contextual Prompting. At each student pre-training step, the frozen teacher encodes the same masked attribute view supplied to the student. The contextual hyperedge prototypes $\mathbf{P}^{\mathrm{ctx}}$ and node-level semantic prompts $\mathbf{Q}$ are recomputed using Eqns. (10) and (11), respectively. Thus, the semantic guidance adapts to the current masked view while the teacher parameters remain fixed. The teacher forward pass is performed without gradient tracking, and the resulting contextual prototypes are detached from the computation graph.

Adaptive Fusion and Gradient Flow. In Eqn. (18), stop-gradient is applied to the gate input $\mathbf{u}_i$ and the reliability-discrepancy score η_i. This prevents gradients from propagating back to the graph-student representations through the gate-input and discrepancy paths, while the gate parameters ω remain trainable through the fused reconstruction objective. The graph student continues to receive gradients through its direct contribution to the fused representation and through the graph-specific reconstruction objective. The frozen teacher receives no gradients during student pre-training.

A.3 Algorithm and Complexity Analysis

A.3.1 Algorithms

The main procedures of our proposed HyperRoGMAE are summarized in Algorithm 1.

Algorithm 1 Training and Inference of HyperRoGMAE

Input: Adjacency matrix $\mathbf{A}_o$, node attributes $\mathbf{X}$, semantic scales $\mathcal{K} = \{K_1, \ldots, K_S\}$, memberships per scale κ, attribute mask ratio ρ_m, hyperedge-aware masking mixture coefficient ρ_h, number of perturbed views T, teacher and student training epochs T_H, T_G, and inference adjacency matrix $\mathbf{A}_o^{\text{eval}}$.
Output: Robust node representations $\mathbf{Z}_{\text{eval}}$.

#Preprocessing: Attribute-Induced Semantic Reference
Construct normalized semantic representations $\mathbf{Y}$ from $\mathbf{X}$, applying truncated SVD when applicable.
for $s = 1, \ldots, S$ **do**
  Obtain K_s cluster centers $\{\mathbf{c}_e^{(s)}\}_{e=1}^{K_s}$ by mini-batch k-means on $\mathbf{Y}$.
  Normalize the centers, compute affinities, and construct $\mathbf{H}^{(s)}$ by retaining the top-κ memberships per node.
  Generate T independently perturbed attribute views and apply the same fitted projection, when applicable, followed by row-wise normalization.
  Compute assignment margins $\Delta_i^{(s)}$ and stability scores $q_i^{(s)}$ using the fixed cluster centers at scale s.
end for
Concatenate $\mathbf{H} = [\mathbf{H}^{(1)} \mid \cdots \mid \mathbf{H}^{(S)}]$ and compute $\mathbf{D}_v$ and $\mathbf{D}_e$.
Calculate c_i and construct $\mathbf{H}_c$ and $\mathbf{D}_{e,c}$.
Construct $\mathbf{W}_o$ from $\mathbf{A}_o$, $\mathbf{X}$, and $\mathbf{H}$ and normalize it to obtain $\widehat{\mathbf{W}}_o$.
#Hypergraph Teacher Pre-Training
for $t = 1, \ldots, T_H$ **do**
  Sample $\mathcal{V}_{\text{mask}}^H$ using hybrid masking with ρ_m and ρ_h, and construct $\widetilde{\mathbf{X}}$ using the learnable mask token.
  Compute $\mathbf{Z}_H = f_H(\widetilde{\mathbf{X}}, \mathbf{H})$ and $\widehat{\mathbf{X}}_H = D_H(\mathbf{Z}_H)$.
  Compute $\mathcal{L}_{\text{tea}} = \mathcal{L}_{\text{sce}}(\widehat{\mathbf{X}}_H, \mathbf{X}; \mathcal{V}_{\text{mask}}^H)$.
  Update f_H, D_H, and the learnable mask token by minimizing $\mathcal{L}_{\text{tea}}$.
end for
Restore the selected teacher checkpoint, freeze the teacher, and place it in evaluation mode.
#Graph Student Pre-Training
for $t = 1, \ldots, T_G$ **do**
  Sample $\mathcal{V}_{\text{mask}}$ using hybrid masking with ρ_m and ρ_h, and construct the shared masked attribute $\widetilde{\mathbf{X}}$.
  Compute $\mathbf{Z}_H^{\text{ctx}}$, $\mathbf{P}^{\text{ctx}}$, $\mathbf{Q}$, and $\mathbf{Z}_G$.
  Form semantic guidance $\mathbf{h}_i^{\text{sem}}$, Construct $\mathbf{u}_i$.
  Compute discrepancy d_i and set η_i.
  Compute β_i, with stop-gradient applied to $\mathbf{u}_i$ and η_i.
  Compute $\mathcal{L}_{\text{fused}}$ and $\mathcal{L}_{\text{graph}}$, and minimize $\mathcal{L}_{\text{stu}} = \mathcal{L}_{\text{fused}} + \lambda_{\text{recon}}\mathcal{L}_{\text{graph}}$ to update f_G, D_F, D_G, g_ω, the fusion layer normalization, and the learnable mask token, while keeping the teacher frozen.
end for
#Downstream Inference
Place the trained graph student in evaluation mode.
Construct the normalized weighted adjacency $\widehat{\mathbf{W}}_o^{\text{eval}}$ from $\mathbf{A}_o^{\text{eval}}$, $\mathbf{X}$, and $\mathbf{H}$ using the same edge-weighting and normalization procedure.
return $\mathbf{Z}_{\text{eval}}$.

A.3.2 Complexity Analysis

The complexity of HyperRoGMAE consists primarily of semantic-prior construction, hypergraph-teacher pre-training, and reliability-weighted graph-student pre-training. Let $E = |\mathcal{E}_o|$ denote the number of observed edges and $R_H = \text{nnz}(\mathbf{H}) \leq N\kappa S$ denote the number of nonzero entries in the sparse incidence matrix, where each node retains at most κ memberships at each of the S semantic scales. We use d' and d to denote the semantic-space dimension and hidden representation dimension, respectively.

Time Complexity. Treating the encoder depths and the numbers of SVD and clustering iterations as constants, truncated SVD requires $\mathcal{O}(NDd')$, while multi-scale mini-batch k-means requires $\mathcal{O}(Nd'M)$, where $M = \sum_{s=1}^{S} K_s$ is the total number of semantic hyperedges. Evaluating assignment stability over T perturbed attribute views costs $\mathcal{O}(TNDd' + TNd'M)$. Computing the reliability weights of the observed edges requires $\mathcal{O}(ED + E\kappa S)$, where the two terms correspond to attribute similarity and sparse hypergraph-membership consistency, respectively. The semantic prior remains fixed after construction, and the edge weights can be computed once and cached for a fixed graph topology. One epoch of hypergraph-teacher pre-training requires $\mathcal{O}(R_H(D + d) + Nd(D + d))$, including sparse node-hyperedge propagation, feature transforma-

tion, and attribute decoding. The frozen teacher and contextual hyperedge prompting require $\mathcal{O}(R_H(D+d)+Nd^2)$, while the graph student requires $\mathcal{O}(NDd+Nd^2+Ed)$. Combining the two pre-training stages and treating the numbers of teacher and student epochs as fixed constants, the overall training complexity, excluding semantic-prior preprocessing and assuming cached edge weights, is $\mathcal{O}(R_H(D+d)+Nd(D+d)+Ed)$. Because the sparse incidence matrix satisfies $R_H \leq N\kappa S$, this complexity is bounded by $\mathcal{O}(N\kappa S(D+d)+Nd(D+d)+Ed)$.

A.4 Experimental Settings

A.4.1 Datasets

Our evaluation includes five attributed graph datasets used in the main text and two additional industrial control-system (ICS) datasets studied in this appendix. The former includes three citation networks (Cora, Citeseer, and PubMed) and two Amazon co-purchase networks (Photo and Computers). The supplementary experiments utilize HAI 23.05 and SWaT A1 to evaluate the applicability of HyperRoGMAE under industrial anomaly detection.

Attributed Graph Datasets. The five datasets are utilized to evaluate node classification performance under the structural attacks. For each dataset, node attributes and train/validation/test splits are kept fixed across global structural attack settings. For Photo and Computers, we generate reproducible 10%/10%/80% train/validation/test node splits using the experiment seed.

Industrial Control-System Datasets. HAI 23.05 and SWaT A1 are used for supplementary experiments under an anomaly detection protocol. Each sliding temporal window is converted into an attributed graph whose nodes represent process variables. Both the functional graph and the semantic hypergraph are constructed using training data. The detection threshold is selected using a held-out normal validation set, and test anomaly labels are used only for final evaluation. These experiments assess industrial anomaly detection, a task distinct from the structural-attack node classification evaluation on the static graph datasets.

Table 4: Statistics of attributed graph datasets and industrial control-system datasets.

(a) Attributed graph datasets

Domain	Dataset	Nodes	Edges	Features	Classes	Train	Val.	Test
Citation	Cora	2,485	5,069	1,433	7	247	249	1,988
	Citeseer	2,110	3,668	3,703	6	210	211	1,688
	PubMed	19,717	44,324	500	3	1,971	1,972	15,774
Amazon	Photo	7,650	119,081	745	8	765	765	6,120
	Computers	13,752	245,861	767	10	1,375	1,375	11,002

(b) Industrial control-system datasets

Dataset	Nodes per graph	Train	Val.	Test	Test anomalies	Anomaly ratio
HAI 23.05	86	698,400	198,000	284,400	11,384	4.00%
SWaT A1	51	396,000	99,000	449,919	54,621	12.14%

Note: For ICS datasets, split sizes refer to time points before sliding-window construction. Each window forms a graph with process variables as nodes. Test anomalies and Anomaly ratio report the count and percentage of anomalous test time points, respectively.

A.4.2 Baselines

To validate the effectiveness of our proposed HyperRoGMAE, we compare it with representative generative and robust graph masked autoencoders.

- **GraphMAE** (Hou et al., 2022): It is a generative self-supervised model that reconstructs masked node features. It introduces scaled cosine error as the reconstruction criterion and employs a re-mask decoding strategy to learn node representations.
- **MaskGAE** (Li et al., 2023): It is a structure-masked graph autoencoder that learns node representations by reconstructing masked edges from a partially observed graph.

- **Bandana** (Zhao et al., 2024): It is a topology-oriented masked graph learning method that applies bandwidth-aware masking and layer-wise bandwidth prediction to capture structural information at different receptive-field ranges.
- **GCMAE** (Wang et al., 2024b): It integrates graph contrastive learning with masked autoencoding, jointly exploiting local reconstruction signals and global representation consistency.
- **AUGMAE** (Wang et al., 2024a): It combines masked graph autoencoding with adversarial masking and alignment–uniformity regularization to learn generalizable node representations from easy-to-hard masked views.
- **ProtoMGAE** (Zheng & Jia, 2024): It introduces prototype-based semantic guidance into masked graph autoencoding, using global prototype information to assist masked attribute reconstruction and representation learning.
- **AMGAE** (Hu et al., 2025): It formulates masked reconstruction as an adversarial learning problem, where a discriminator distinguishes reconstructed features from original features to improve representation robustness.
- **DGMAE** (Zheng et al., 2025): It is a discrepancy-aware graph masked autoencoder for heterophilic graphs. It reconstructs discrepancy information between connected nodes instead of relying solely on neighborhood similarity.
- **ArmorGAE** (Zhang et al., 2026): It is a robust graph masked autoencoder designed to reduce the influence of perturbed topology during masked representation learning.

For the supplementary industrial experiments, we include recurrent and temporal-convolutional autoencoders together to compare temporal reconstruction and graph-based modeling under the same normal-only training protocol.

- **LSTM-AE** (Malhotra et al., 2016): It is a recurrent autoencoder that uses long short-term memory networks to reconstruct normal multivariate time series. Anomalies are identified according to reconstruction errors.
- **TCN-AE** (Thill et al., 2021): It is a temporal-convolutional autoencoder that models normal temporal dynamics with dilated causal convolutions and detects anomalous observations through reconstruction errors.

A.4.3 Graph Attack Settings

We consider four structural attack methods: Metattack, Nettack, Random edge insertion, and DICE. For Cora, Citeseer, and PubMed, we load pre-generated Metattack and Nettack adjacency matrices from the data released by DeepRobust and ProGNN, along with the corresponding data splits and target-node annotations. For Photo and Computers, we load locally stored Metattack adjacency matrices under the evaluated perturbation ratios. Metattack is a global structural attack evaluated at perturbation ratios of $\{5\%, 10\%, 15\%, 20\%, 25\%\}$. Nettack is a targeted structural attack evaluated with ratios of $\{1, 2, 3, 4, 5\}$ edge modifications per target node. Random edge insertion and DICE are evaluated at perturbation ratios of $\{10\%, 20\%, 30\%, 40\%, 50\%\}$.

A.4.4 Training and Evaluation Protocol

Self-Supervised Pre-training. Following standard graph masked autoencoders, the semantic teacher is optimized for 200 epochs and restored from the checkpoint with the lowest masked reconstruction loss, after which its parameters are frozen. The graph student is then optimized for at most 300 epochs. Student checkpoint selection uses an exponential moving average of the training objective. Early stopping is activated after 100 epochs and is triggered if the smoothed objective fails to improve by at least 10^{-4} for 80 consecutive epochs.

Linear probing. Following the standard linear probing evaluation protocol in self-supervised graph representation learning (e.g., ArmorGAE and GraphMAE), we extract node embeddings from the trained graph student and apply row-wise ℓ_2 normalization. The representation encoder is strictly frozen throughout the entire downstream evaluation. A single linear classification layer is attached

and trained for 500 epochs on the labeled training nodes using the Adam optimizer with a learning rate of 10^{-2}, weight decay of 0, and a batch size of 512. The classifier checkpoint is selected based on the highest classification accuracy on the validation split, and the corresponding test node accuracy is reported.

A.4.5 Hyperparameter Search and Final Configurations.

Search protocol. Hyperparameters are selected according to validation performance. We search the edge-reliability mixture over $\{0, 0.125, 0.25, 0.5, 0.75, 1\}$, feature-mask rate over $\{0.3, 0.4, 0.5, 0.6, 0.7\}$, hyperedge-mask mixture over $\{0, 0.25, 0.5, 0.8, 1\}$, graph-reconstruction coefficient over $\{0, 0.25, 0.5, 1, 2\}$, hidden dimension over $\{128, 256, 512\}$, and semantic hyperedge scales among $\{128\}$, $\{64, 128\}$, $\{32, 64, 128\}$, and $\{16, 32, 64, 128\}$. Once selected, the configurations are fixed across the corresponding clean, poisoning, and evasion evaluations.

Final Configurations. Unless otherwise specified, we retain $\kappa = 2$ semantic hyperedge memberships per node at each scale and set the soft-incidence temperature to $\tau_h = 0.2$. The reliability concentration exponent and residual edge-weight floor are set to $\gamma = 1.0$ and $\epsilon_e = 0$, respectively. For the semantic guidance branch, we use $\lambda_h = 0.8$ to combine the contextual teacher representation and the node-level semantic prompt, and set the maximum semantic fusion strength to $s_f = 1.0$. The attribute mask ratio and hyperedge-aware masking mixture coefficient are fixed to $\rho_m = 0.5$ and $\rho_h = 0.8$, respectively. The hypergraph teacher contains two message-passing layers on all datasets. The graph student contains one layer on Cora, Citeseer, and PubMed and two layers on Photo and Computers. For the Random and DICE experiments on Cora, we use $\epsilon_e = 0.25$; all other reported experiments use $\epsilon_e = 0$.

A.5 Additional Experimental Results

A.5.1 Results of the Additional Robustness Study

Results under Metattack. Table 5 reports node classification accuracy under Metattack in poisoning and evasion scenarios. Averaged over the ten attacked settings, HyperRoGMAE achieves the highest accuracy on four of the five datasets, outperforming the strongest baseline by 10.02%, 3.81%, and 5.61% on Cora, Citeseer, and PubMed, respectively. Its advantage becomes particularly pronounced under severe structural corruption: at a poisoning ratio of 25%, the corresponding improvements are 26.31%, 8.61%, and 6.24%. HyperRoGMAE also ranks first in every attacked setting on Photo and in every evasion scenario on Computers, although ArmorGAE remains stronger under poisoning on Computers. These results demonstrate robustness across both attack scenarios, with particularly substantial benefits at high perturbation ratios.

The comparison with graph masked autoencoder baselines highlights the ability of HyperRoGMAE to preserve predictive information as structural corruption intensifies. On Cora, increasing the poisoning ratio from 5% to 25% reduces the accuracy of MaskGAE and ArmorGAE from 80.65% and 81.25% to 52.05% and 52.28%, respectively, whereas HyperRoGMAE decreases only from 80.46% to 78.59%. Thus, although these baselines perform better at the smallest poisoning setting, their advantage does not persist under stronger attacks. This pattern is consistent with the motivation behind HyperRoGMAE: its attribute-induced semantic hypergraph provides higher-order context beyond the observed graph neighborhoods, while semantic guidance and edge-reliability weighting help reduce the influence of corrupted connections. The resulting complementary semantic reference can support reconstruction when graph-based context becomes unreliable, offering an advantage over relying primarily on the observed topology for masked representation learning.

Results under Nettack. Table 6 reports node classification accuracy under Nettack in poisoning and evasion scenarios. Averaged over the ten attacked settings, HyperRoGMAE achieves the best performance on all three citation datasets, with accuracies of 77.16%, 79.95%, and 86.46% on Cora, Citeseer, and PubMed, exceeding the strongest competing averages by 3.20%, 5.96%, and 3.13%, respectively. Its advantage is particularly evident under larger poisoning ratios. With five edge modifications per target node, HyperRoGMAE achieves 73.49%, 80.00%, and 84.30%, outperforming the strongest competing methods by 6.57%, 10.79%, and 5.93%, respectively. Under evasion, it ranks first at every ratio on PubMed and at the largest ratio on all three datasets. Nevertheless, the

Table 5: Node classification accuracy (%) under Metattack in poisoning and evasion scenarios. The highest and second-highest mean accuracies for each dataset and attack setting are highlighted in **bold** and underlined, respectively.

Dataset	Methods	Poisoning (Acc.)					Evasion (Acc.)				
		5%	10%	15%	20%	25%	5%	10%	15%	20%	25%
Cora	GraphMAE'22	79.07±0.39	71.66±0.52	66.88±0.21	53.38±0.58	49.44±0.75	80.12±0.09	72.94±0.13	71.68±0.09	63.23±0.18	53.92±0.13
	MaskGAE'23	80.65±0.64	75.67±0.33	67.87±0.32	54.26±1.17	52.05±0.55	65.36±0.84	60.90±0.83	51.54±1.23	45.81±1.05	41.71±1.35
	Bandana'24	79.74±1.15	75.45±1.08	68.29±0.85	55.57±0.92	51.63±0.88	79.22±0.98	74.01±0.75	67.45±1.12	53.67±1.04	49.45±1.31
	AUGMAE'24	77.73±0.35	72.30±0.33	65.10±0.31	54.72±0.45	48.80±0.38	80.35±1.00	78.03±0.29	73.27±0.28	66.08±0.35	60.76±0.57
	GCMAE'24	78.55±0.60	73.02±0.35	66.61±0.59	55.50±0.57	48.85±0.57	81.41±0.20	77.87±0.71	73.99±0.27	66.16±0.56	58.95±0.37
	ProtoMGAE'24	77.86±0.16	69.44±0.47	62.97±0.56	52.56±0.84	47.39±1.04	78.80±0.55	70.75±0.59	61.30±0.98	48.94±0.61	44.95±0.57
	AMGAE'25	77.64±0.81	72.19±0.61	65.02±0.64	53.33±0.81	47.49±0.82	81.08±0.44	77.76±0.57	74.09±0.77	65.59±0.66	58.60±0.97
	DGMAE'25	78.48±0.16	74.08±0.11	68.31±0.15	57.85±0.10	50.68±0.19	81.05±0.11	77.45±0.13	74.12±0.17	64.70±0.14	58.47±0.12
	ArmorGAE'26	**81.25±0.33**	77.16±0.51	71.37±0.45	59.62±1.54	52.28±1.37	**81.87±0.23**	77.16±0.49	75.16±1.35	60.81±1.47	57.79±0.44
	HyperRoGMAE	80.46±0.11	**78.78±1.36**	**78.85±0.31**	**78.91±0.23**	**78.59±0.72**	80.06±0.47	**78.98±0.41**	**79.95±0.13**	**80.36±0.28**	**79.77±0.09**
Citeseer	GraphMAE'22	68.85±0.32	67.90±0.36	62.59±0.38	54.44±0.42	53.95±0.82	69.75±0.09	68.84±0.09	65.52±0.29	63.39±0.22	61.61±0.09
	MaskGAE'23	74.38±0.43	72.35±0.51	71.46±0.77	63.63±1.24	62.69±0.58	73.31±0.73	70.11±1.01	70.28±0.60	55.91±1.78	64.77±0.81
	Bandana'24	73.13±1.31	71.90±0.50	69.60±0.81	63.93±0.57	64.15±1.61	72.69±0.67	71.30±0.98	69.44±0.90	60.02±0.96	62.07±2.51
	AUGMAE'24	71.47±0.20	66.02±0.41	66.18±0.36	57.61±0.23	56.62±0.33	72.59±0.16	71.74±0.17	71.02±0.29	66.85±0.32	66.72±0.27
	GCMAE'24	67.89±1.03	61.38±0.72	57.65±1.39	50.77±1.00	45.37±0.69	68.47±0.65	67.11±1.15	67.57±0.75	64.38±1.11	65.68±0.32
	ProtoMGAE'24	73.36±0.36	69.12±0.41	68.60±0.55	57.99±0.69	60.34±0.79	72.53±0.69	70.68±0.59	68.00±1.61	60.84±1.03	62.84±1.76
	AMGAE'25	71.71±0.33	66.71±0.71	65.65±0.76	56.94±0.26	56.21±0.91	73.16±0.28	71.99±0.42	70.95±0.84	66.59±0.92	66.86±0.55
	DGMAE'25	72.93±0.00	71.27±0.00	69.37±0.00	57.70±0.00	59.60±0.00	73.82±0.00	73.10±0.00	70.26±0.00	64.16±0.00	65.88±0.00
	ArmorGAE'26	**75.59±0.14**	73.50±0.12	71.94±0.32	66.04±0.43	64.14±1.65	**74.97±0.10**	73.02±0.70	72.32±0.36	64.78±0.63	66.18±0.58
	HyperRoGMAE	74.96±0.27	**73.78±0.08**	**74.19±0.20**	**72.03±0.18**	**72.76±0.53**	74.89±0.11	**74.47±0.15**	**74.85±0.22**	**74.00±0.17**	**74.66±0.15**
PubMed	GraphMAE'22	70.27±0.08	73.80±0.47	69.24±0.38	65.76±0.26	59.61±0.60	80.86±0.16	78.94±0.12	76.44±0.19	74.04±0.32	71.05±0.36
	MaskGAE'23	80.55±0.03	77.32±0.02	75.64±0.09	75.37±0.08	72.89±0.08	81.60±0.08	78.48±0.04	76.10±0.06	74.23±0.07	72.28±0.04
	Bandana'24	82.24±0.08	79.63±0.08	77.51±0.06	74.83±0.11	72.63±0.07	80.90±0.09	78.25±0.18	75.90±0.20	73.46±0.18	71.60±0.07
	AUGMAE'24	80.59±0.27	79.60±0.01	77.81±0.35	76.39±0.69	74.07±0.62	81.00±0.02	78.93±0.02	76.30±0.03	73.60±0.10	71.13±0.04
	GCMAE'24	76.55±0.45	72.86±0.66	67.38±1.04	62.15±0.37	59.45±0.72	75.80±0.21	72.02±0.41	67.89±0.37	63.31±0.43	60.08±0.53
	ProtoMGAE'24	82.00±0.13	80.78±0.25	80.02±0.17	78.53±0.16	76.23±0.44	81.39±0.18	79.86±0.25	77.87±0.37	75.87±0.30	73.33±0.40
	AMGAE'25	81.67±0.18	80.02±0.15	78.53±0.31	77.04±0.10	75.06±0.39	81.82±0.08	80.01±0.08	78.07±0.17	76.05±0.09	73.70±0.13
	DGMAE'25	82.68±0.03	81.73±0.05	81.55±0.05	79.71±0.06	77.85±0.04	79.83±0.12	79.83±0.12	77.70±0.20	73.90±0.03	71.30±0.03
	ArmorGAE'26	82.14±0.06	80.93±0.09	78.73±0.21	77.56±0.11	76.08±0.15	81.73±0.09	79.38±0.13	76.93±0.14	73.91±0.11	72.27±0.14
	HyperRoGMAE	**84.57±0.06**	**82.44±0.03**	**84.27±0.16**	**84.24±0.09**	**84.09±0.04**	**84.61±0.05**	**84.57±0.03**	**84.55±0.04**	**84.48±0.02**	**84.34±0.06**
Photo	GraphMAE'22	69.39±17.32	77.93±8.17	80.98±1.72	80.52±1.96	79.77±2.58	79.28±12.83	77.99±13.28	76.78±9.89	77.77±10.35	77.09±11.55
	MaskGAE'23	90.10±0.15	89.76±0.14	89.00±0.22	88.52±0.27	88.00±0.25	90.25±0.18	89.75±0.12	89.47±0.21	89.20±0.13	89.08±0.15
	Bandana'24	90.23±0.25	88.69±0.09	88.58±0.14	88.63±0.12	88.42±0.19	90.25±0.16	89.46±0.27	88.41±0.14	88.89±0.21	89.01±0.13
	AUGMAE'24	90.49±0.28	89.63±0.39	89.02±0.24	88.23±0.82	88.15±0.51	90.63±0.35	89.79±0.47	89.72±0.40	89.02±0.52	88.85±0.49
	GCMAE'24	89.18±1.13	88.68±0.72	87.56±2.39	87.04±2.87	84.54±0.75	87.48±5.48	86.84±6.75	85.53±7.11	85.05±6.81	84.67±6.50
	ProtoMGAE'24	90.26±0.15	89.64±0.56	89.70±0.48	89.43±0.27	89.63±0.31	90.46±0.21	90.10±0.20	90.24±0.50	89.91±0.41	89.94±0.36
	DGMAE'25	90.47±0.60	89.81±0.79	89.35±0.88	88.76±1.27	88.49±1.38	90.23±0.50	89.39±0.40	89.23±0.56	88.83±0.46	88.61±0.51
	AMGAE'25	87.93±1.54	87.15±1.76	86.60±1.39	87.89±1.79	87.07±2.39	88.29±3.96	87.25±4.43	87.39±4.30	87.28±3.71	87.02±3.50
	ArmorGAE'26	89.77±0.60	90.07±0.23	89.33±0.32	89.17±0.38	89.11±0.26	90.34±0.31	89.87±0.26	89.69±0.24	89.47±0.25	88.93±0.27
	HyperRoGMAE	**92.33±0.09**	**92.31±0.23**	**92.29±0.07**	**91.98±0.07**	**92.24±0.09**	**92.78±0.14**	**92.37±0.10**	**92.61±0.03**	**92.34±0.10**	**92.42±0.06**
Computers	GraphMAE'22	64.72±6.33	65.94±4.02	64.78±3.74	62.04±3.51	61.60±3.24	70.10±1.95	68.63±1.78	67.87±1.74	67.21±1.83	66.02±1.23
	MaskGAE'23	86.48±0.15	86.40±0.14	86.00±0.09	85.56±0.10	84.87±0.09	86.32±0.12	85.77±0.18	85.51±0.19	85.00±0.26	84.83±0.22
	Bandana'24	86.04±0.09	85.69±0.19	85.48±0.21	84.60±0.25	85.80±0.17	86.23±0.25	85.69±0.09	85.58±0.14	84.63±0.12	84.42±0.19
	AUGMAE'24	85.37±0.26	83.28±0.42	77.93±0.47	75.97±0.36	72.44±0.92	84.95±0.37	83.98±0.18	83.62±0.28	83.06±0.89	82.54±0.41
	GCMAE'24	87.36±0.38	86.05±0.70	85.64±0.39	84.90±0.74	84.45±0.57	87.27±0.56	86.43±0.42	86.19±0.31	85.18±0.23	84.95±0.40
	ProtoMGAE'24	87.49±0.34	86.61±0.37	86.32±0.22	85.63±0.31	85.70±0.16	87.28±0.30	86.61±0.33	86.24±0.32	86.00±0.22	85.53±0.24
	DGMAE'25	84.62±1.72	81.84±3.62	82.98±1.56	82.78±0.58	80.79±2.75	85.08±0.55	84.01±0.45	83.37±0.42	82.77±0.39	82.23±0.55
	AMGAE'25	86.78±0.58	85.93±0.29	84.87±0.11	84.57±0.46	84.22±0.40	86.87±0.22	85.83±0.28	85.04±0.27	84.83±0.39	84.17±0.43
	ArmorGAE'26	**89.35±0.13**	**88.17±0.04**	**88.38±0.27**	**88.63±0.14**	**88.25±0.16**	86.66±0.31	86.08±0.33	85.49±0.28	85.34±0.21	85.09±0.30
	HyperRoGMAE	87.27±0.12	86.79±0.12	86.76±0.04	86.60±0.12	86.62±0.08	**87.36±0.03**	**87.10±0.03**	**87.01±0.06**	**86.32±0.02**	**86.50±0.06**

advantage is not uniform across settings: MaskGAE performs better at intermediate evasion ratios on Citeseer, while several baselines remain competitive at the smallest ratio.

Compared with the evaluated GMAE methods, a key advantage of HyperRoGMAE is its ability to preserve predictive information as targeted neighborhood corruption intensifies. On Citeseer, increasing the poisoning perturbation ratio from one to five reduces the accuracy of ArmorGAE from 81.79% to 58.25%, whereas HyperRoGMAE decreases only from 80.95% to 80.00%. MaskGAE likewise decreases from 81.43% to 49.05% under poisoning, despite retaining strong evasion performance on this dataset. These observations highlight that robustness in one attack scenario does not necessarily extend to the other. They are consistent with the motivation behind HyperRoGMAE: attribute-induced semantic hyperedges provide higher-order context beyond the attacked graph neighborhoods, while semantic guidance and edge-reliability weighting help limit the influence of unreliable connections. This complementary semantic reference can support target-node

Table 6: Node classification accuracy (%) under Nettack in poisoning and evasion scenarios. The highest and second-highest distinct mean accuracies for each dataset and attack setting are highlighted in **bold** and underlined, respectively.

Dataset	Methods	Poisoning (Acc.)					Evasion (Acc.)				
		1	2	3	4	5	1	2	3	4	5
Cora	GraphMAE'22	76.02±0.38	70.26±0.57	67.10±0.98	67.38±1.34	58.80±1.51	77.11±0.01	74.70±0.03	74.85±0.06	68.67±0.02	63.86±0.01
	MaskGAE'23	78.52±1.08	70.12±1.60	68.19±1.80	61.93±1.10	62.41±1.30	68.39±1.46	63.28±1.62	59.88±2.05	57.56±1.35	48.70±1.58
	Bandana'24	77.83±2.03	75.06±1.71	72.29±1.43	59.76±1.88	55.06±1.62	80.12±1.43	77.35±0.72	75.06±0.94	67.35±1.14	62.29±1.62
	AUGMAE'24	79.36±1.30	74.36±1.46	71.69±0.78	66.77±1.78	63.98±0.98	81.45±0.59	78.41±0.73	76.11±0.84	72.28±0.62	69.88±0.93
	GCMAE'24	79.28±1.01	75.66±2.32	73.25±1.32	66.51±1.01	64.82±1.57	80.72±0.85	79.03±1.08	**76.87±1.01**	72.05±1.57	71.32±1.57
	ProtoMGAE'24	79.66±0.80	77.18±1.22	72.26±1.47	65.77±0.65	63.39±1.34	79.06±1.08	76.44±0.64	74.96±0.92	72.05±0.55	70.90±0.94
	AMGAE'25	78.80±1.62	76.14±1.57	74.94±1.32	67.71±2.15	63.37±1.08	80.24±1.83	77.59±1.83	**76.87±2.15**	72.53±1.98	71.08±1.91
	DGMAE'25	76.89±1.02	75.78±0.88	72.26±0.62	67.35±0.12	66.92±0.74	80.12±0.97	76.29±0.87	73.25±0.73	72.52±0.93	68.07±0.63
	ArmorGAE'26	80.12±1.08	75.06±1.00	73.37±0.59	63.73±1.37	62.17±1.80	**83.98±0.18**	77.87±1.43	75.18±1.37	70.00±1.66	66.63±1.21
	HyperRoGMAE	**81.20±0.59**	**78.55±0.90**	**75.90±2.02**	**74.46±1.77**	**73.49±0.00**	82.17±0.48	**80.00±0.96**	75.90±1.32	**75.42±0.59**	**74.46±1.18**
Citeseer	GraphMAE'22	69.84±1.04	63.65±1.50	49.21±4.80	38.73±2.40	44.92±5.60	69.36±0.78	55.70±2.51	57.91±1.09	50.00±1.85	42.61±1.37
	MaskGAE'23	81.43±0.73	73.65±4.67	61.90±2.75	59.52±2.58	49.05±2.70	82.44±0.30	**82.26±0.50**	**81.69±1.30**	**80.32±1.80**	79.18±2.40
	Bandana'24	80.63±0.63	68.73±6.96	61.59±6.34	56.51±3.11	47.62±5.26	81.82±0.68	80.05±1.15	79.32±1.63	76.66±2.14	74.43±2.97
	AUGMAE'24	80.95±0.00	77.46±1.07	65.34±1.62	58.73±1.63	60.48±3.97	79.37±0.00	77.41±0.98	65.24±0.50	64.90±0.50	63.49±0.00
	GCMAE'24	76.51±1.74	71.75±0.71	67.62±4.71	62.54±0.87	69.21±1.81	79.05±1.74	75.24±1.42	74.60±3.72	71.75±2.07	75.24±2.13
	ProtoMGAE'24	81.27±1.33	71.49±3.26	59.37±2.40	57.11±1.12	49.95±2.51	73.59±5.04	66.43±3.11	56.55±3.35	49.70±2.51	40.59±4.71
	AMGAE'25	79.69±0.71	76.19±3.18	65.08±4.05	60.95±2.41	56.51±1.42	79.37±0.00	69.52±4.94	64.13±0.87	63.81±0.71	63.49±0.00
	DGMAE'25	80.95±0.00	74.60±0.00	61.90±0.00	58.73±0.00	55.56±0.00	80.95±0.00	79.37±0.00	66.67±0.00	63.49±0.00	58.73±0.00
	ArmorGAE'26	**81.79±0.48**	**80.63±0.85**	64.92±0.27	61.27±1.90	58.25±2.66	**82.86±0.00**	80.32±0.00	79.37±0.73	75.87±1.38	74.60±2.13
	HyperRoGMAE	80.95±0.00	**80.63±0.63**	**80.78±0.00**	**76.51±0.63**	**80.00±0.78**	80.95±0.00	80.32±0.78	80.63±0.63	79.37±0.00	**79.37±0.00**
PubMed	GraphMAE'22	83.87±0.00	82.80±0.00	79.03±0.00	74.73±0.00	70.43±0.00	85.99±0.45	85.59±0.59	84.73±0.30	83.12±0.29	79.89±1.64
	MaskGAE'23	82.80±0.00	77.96±0.00	70.91±0.29	66.56±0.67	58.71±0.32	82.96±0.42	78.55±0.38	73.98±0.26	68.49±0.26	62.74±0.34
	Bandana'24	84.78±0.59	80.59±0.66	75.75±0.78	67.47±0.77	60.86±1.98	84.68±0.73	78.28±1.03	74.41±1.43	68.44±0.96	61.45±0.68
	AUGMAE'24	85.22±0.26	84.81±0.16	83.33±0.12	80.65±0.00	78.37±0.23	87.10±0.93	85.48±0.76	84.41±0.81	82.53±0.62	80.07±0.51
	GCMAE'24	83.44±0.80	80.54±1.23	77.31±0.88	75.59±1.12	72.47±0.88	84.09±0.81	80.97±0.81	78.92±0.59	76.45±1.39	72.04±0.85
	ProtoMGAE'24	87.19±1.36	83.75±0.46	80.85±0.80	77.97±0.50	75.64±1.58	87.26±0.93	86.04±0.76	84.19±0.98	82.53±0.62	80.07±0.51
	AMGAE'25	**87.39±0.98**	84.77±0.90	82.19±0.48	79.72±1.37	77.74±0.72	87.49±0.62	86.04±0.70	84.38±0.59	82.87±1.14	80.72±0.85
	DGMAE'25	86.48±0.56	84.78±1.33	81.72±0.98	80.11±0.72	76.96±1.98	87.17±0.80	86.83±0.88	83.87±0.61	80.54±0.80	76.56±1.17
	ArmorGAE'26	87.01±0.54	85.17±0.79	80.32±0.92	79.10±0.54	77.92±0.35	87.31±0.02	86.39±0.09	83.35±0.11	82.78±0.14	80.46±0.85
	HyperRoGMAE	87.31±0.26	**87.53±0.22**	**86.77±0.26**	**86.02±0.00**	**84.30±0.53**	**87.63±0.00**	**87.53±0.22**	**87.20±0.22**	**86.67±0.40**	**83.66±0.65**

representations when their observed local structure becomes misleading, reducing dependence on corrupted neighborhoods during masked representation learning.

Anomaly Detection in Industrial Control Systems. Table 7 presents the results of Ours-T, the temporal adaptation of HyperRoGMAE, on two industrial control system datasets. On HAI, Ours-T achieves the highest AUROC of 77.51%, exceeding TCN-AE by 3.90%. Its AP of 40.67% is slightly below TCN-AE's 41.21%, while its F1 score of 46.62% is lower than those of both baselines under the adopted thresholding protocol. On SWaT, Ours-T also achieves the highest AUROC of 83.07%, exceeding LSTM-AE by 0.92%, but does not improve upon its AP or F1 score. These results suggest improved discrimination as measured by AUROC, although the gains do not consistently extend to precision-recall performance or threshold-based detection. This supplementary evaluation supports the potential applicability of the temporal adaptation to ICS anomaly detection.

Table 7: The results of ICS anomaly detection (%).

Dataset	Method	AUROC	AP	Precision	Recall	F1
HAI	LSTM-AE	68.41	38.32	66.67	**37.21**	**47.76**
	TCN-AE	73.61	**41.21**	**70.45**	36.05	47.69
	Ours-T	**77.51**	40.67	65.96	36.05	46.62
SWaT	LSTM-AE	82.15	**71.90**	**18.69**	82.68	**30.49**
	TCN-AE	81.93	71.54	17.00	**89.47**	28.57
	Ours-T	**83.07**	71.05	18.58	82.24	30.32

A.5.2 Results of Additional Ablation Study

We conduct additional ablation experiments on Cora and Citeseer under the clean setting and 25% Metattack in the poisoning and evasion settings. Table 8 reports mean accuracy and standard deviation over three independent pre-training seeds. We compare the full model with eight ablation variants: **(1)** ***w-Attr***: An attribute-only control that retains self-loops but excludes cross-node propaga-

tion and teacher guidance. **(2) *F-Edge***: A graph student using feature-based observed-edge weighting and random node masking. **(3) *FM-Edge***: F-Edge augmented with hypergraph-membership consistency in edge weighting, while retaining random node masking. **(4) *HG-Mask***: FM-Edge with hyperedge-guided node masking in place of random node masking, without teacher guidance. **(5) *T-Fix***: HG-Mask augmented with contextual teacher representations, using a fixed fusion coefficient and no contextual hyperedge prompts. **(6) *TP-Fix***: T-Fix further augmented with contextual hyperedge prompts, while retaining fixed fusion. **(7) *kNN-Prior***: An alternative-prior control using an attribute-based k-nearest-neighbor graph in place of the semantic hypergraph. **(8) *HG-Only***: A hypergraph-only control that uses representations from the hypergraph encoder without the observed-graph student. The full model replaces fixed fusion in *TP-Fix* with the learned node-aware fusion strategy.

Table 8: Ablation results on Cora and Citeseer under the clean setting and 25% Metattack. The best result is highlighted in bold, and the runner-up is underlined.

Variant	Cora			Citeseer		
	Clean	**Poisoning**	**Evasion**	**Clean**	**Poisoning**	**Evasion**
w-Attr	49.96 ± 3.09	49.96 ± 3.09	49.96 ± 3.09	50.55 ± 1.83	50.55 ± 1.83	50.55 ± 1.83
F-Edge	78.08 ± 1.16	73.76 ± 0.90	77.20 ± 1.63	72.65 ± 0.16	70.74 ± 0.76	72.47 ± 0.25
FM-Edge	77.45 ± 1.25	73.64 ± 1.20	76.48 ± 0.40	73.05 ± 0.35	71.98 ± 0.30	73.08 ± 0.34
HG-Mask	$\underline{79.67 \pm 0.33}$	75.67 ± 0.48	77.65 ± 0.79	$\underline{74.92 \pm 0.61}$	$\underline{73.72 \pm 0.30}$	74.78 ± 0.82
T-Fix	79.09 ± 1.24	$\underline{76.38 \pm 1.30}$	$\underline{77.93 \pm 1.41}$	74.54 ± 0.01	$\underline{73.72 \pm 0.46}$	74.38 ± 0.13
TP-Fix	79.05 ± 0.94	76.35 ± 1.26	77.86 ± 1.19	74.77 ± 0.04	$\mathbf{73.87 \pm 0.32}$	74.55 ± 0.08
kNN-Prior	79.15 ± 0.59	75.45 ± 1.41	77.78 ± 1.50	72.74 ± 1.44	69.96 ± 0.50	72.90 ± 1.25
HG-Only	64.59 ± 0.52	64.59 ± 0.52	64.59 ± 0.52	68.01 ± 0.53	68.01 ± 0.53	68.01 ± 0.53
HyperRoGMAE	$\mathbf{83.48 \pm 0.46}$	$\mathbf{78.59 \pm 0.72}$	$\mathbf{79.77 \pm 0.09}$	$\mathbf{75.50 \pm 0.12}$	72.76 ± 0.53	$\mathbf{74.66 \pm 0.15}$

As shown in Table 8, the full model achieves the highest mean accuracy in four of the six settings. On Citeseer, *TP-Fix* performs best under poisoning, whereas *HG-Mask* achieves the highest accuracy under evasion. Hyperedge-guided masking consistently improves upon *FM-Edge*, with gains of 1.17%-2.22% on Cora and 1.70%-1.87% on Citeseer. It improves upon feature-only weighting on Citeseer but yields no improvement on Cora. Teacher fusion and contextual prompting provide smaller or mixed gains. Compared with *TP-Fix*, the full model improves accuracy by 1.91%-4.43% on Cora, while producing changes ranging from -1.11% to $+0.73\%$ on Citeseer. The full model also outperforms *kNN-Prior* across all settings, supporting the usefulness of the semantic hypergraph relative to the tested pairwise attribute prior. The weaker performance of *w-Attr* and *HG-Only* suggests that neither attribute-only nor hypergraph-only representations are sufficient under the evaluated configurations. Overall, these results show consistent benefits from hyperedge-guided masking, while the gains from teacher-based guidance and learned representations should be interpreted in light of the reported variability.

A.5.3 Reliability Validation

Edge-weight attenuation. Figure 6 compares the weights assigned by Eqn. (13) to retained clean edges and Metattack-injected edges under a 25% perturbation ratio. Across Cora, Citeseer, and PubMed, injected edges tend to receive low weights, whereas retained clean edges concentrate at higher values. The separation is visually most pronounced on PubMed. These distributions support the intended role of the weighting mechanism in preferentially attenuating injected edges. However, the distributions overlap, indicating that some retained edges are also attenuated and some injected edges retain non-negligible weights. The mechanism therefore provides soft attenuation rather than perfect separation. Since w_{ij} is constructed from attribute agreement and hypergraph-membership consistency without probability calibration, it should be interpreted as a propagation weight reflecting relative reliability, rather than the probability that an edge is clean.

Node reliability and label homophily. Figure 7 compares label homophily across three approximately equal-sized groups obtained by ranking nodes according to their semantic reliability scores c_i. On all three datasets, average label homophily increases monotonically from the low- to the high-reliability group. The reported standard deviations are small relative to the differences between

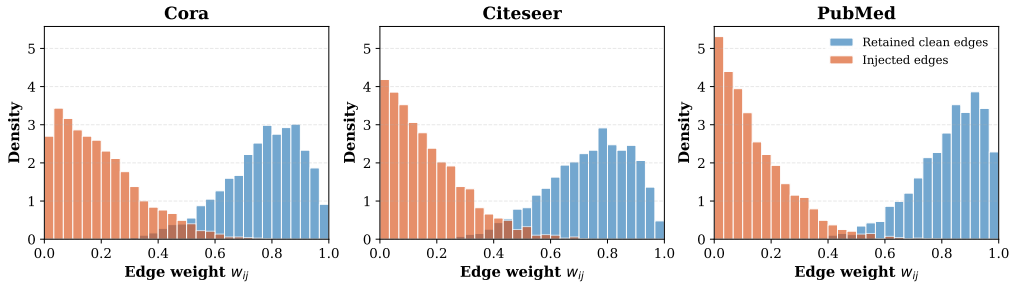


Figure 6: Distributions of edge reliability weights for retained clean edges and Metattack-injected edges on Cora, Citeseer, and PubMed under a 25% perturbation ratio.

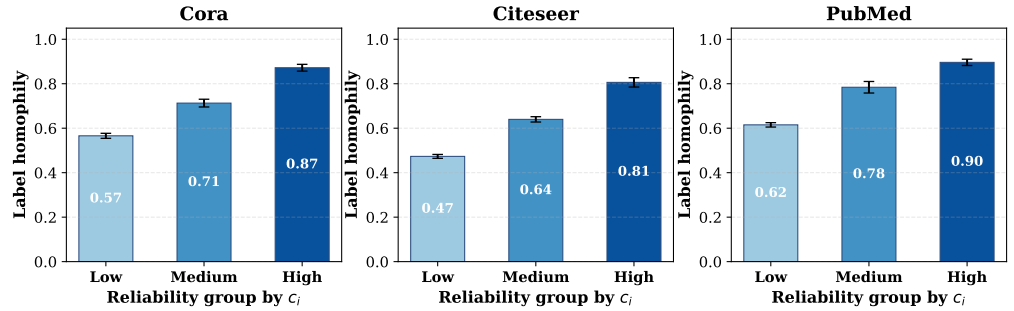


Figure 7: Label homophily across node reliability groups on Cora, Citeseer, and PubMed. Nodes are ranked by c_i and divided into approximately equal-sized low-, medium-, and high-reliability groups. Bars show means, and error bars indicate standard deviations across three independently constructed semantic priors.

group means, indicating limited variation in group-level homophily across the three semantic-prior seeds. These results indicate a positive association between the estimated semantic reliability and label homophily. They do not imply that c_i directly measures semantic correctness: the score is derived from assignment margins and cross-view stability, and label homophily serves only as an external diagnostic of its semantic relevance.

A.6 Discussion and Limitations

There are some limitations. First, HyperRoGMAE focuses on structural attacks and assumes that node attributes remain unchanged. Adversarial attribute perturbations may distort semantic hypergraph construction and reliability estimation, thereby weakening the resulting semantic guidance. Second, the edge-reliability mechanism relies on feature similarity and hypergraph-membership consistency and is therefore more suitable for assortative graphs. On heterophilous graphs, it may assign low weights to legitimate edges connecting nodes with dissimilar attributes or semantic memberships. In the future, we will investigate robust semantic-prior construction under attribute and joint structure-attribute attacks and develop relation-aware semantic construction and edge-weighting mechanisms for heterophilous graphs.