MIRAGE: Hierarchical MI-Surrogate Regulation for Graph Contrastive Learning

Qi Teng, Xue'er Wang*
Zhengzhou University, Sun Yat-sen University
teqi159@gmail.com, wxe0121@hotmail.com

Abstract

Graph contrastive learning (GCL) typically maximizes cross-view agreement via an InfoNCE-based mutual-information (MI) surrogate, applying uniform alignment pressure across all anchors. Yet topology and node attributes vary across a graph in how reliably they support cross-view agreement: strong alignment pressure benefits structurally consistent anchors, whereas it can over-align or degrade boundary, low-agreement, and noisy nodes. Addressing this calibration problem in contrastive node representation learning on small-to-medium attributed graphs, we propose MIRAGE, a hierarchical MI-surrogate regulation framework that couples two core mechanisms: (i) *anchor-wise MI setpoints*, which assign bounded alignment budgets to individual nodes, and (ii) *dual-view MI stabilization*, which anchors branch-level surrogate estimates near a designated target while curbing excessive across-anchor dispersion. Complementing this regulation core, a structure-reliability-guided hypergraph path provides conditional compensation for weak anchors by adaptively gating higher-order aggregation via global and local consistency signals. Across six node-classification benchmarks, MIRAGE achieves highly competitive accuracy—reaching 85.21% on Cora, 73.94% on Citeseer, and 92.57% on ACM. Beyond final accuracy, mechanism-level analyses verify that MIRAGE faithfully tracks prescribed MI-surrogate targets throughout training and that varying the target level induces measurable shifts in downstream performance, establishing target-tracked MI-surrogate regulation as a practical optimization primitive for calibrating cross-view agreement in GCL.

1 Introduction

Graph representation learning has become a foundational tool for relational data in social networks Xiong et al. [2025], Ni et al. [2024], recommender systems Zhang et al. [2024a], Yang et al. [2024b], and molecular discovery Zhou et al. [2024], Feng et al. [2024]. Because labels are often scarce, self-supervised graph contrastive learning (GCL) has emerged as a common paradigm for training encoders by aligning representations from multiple graph views Xu et al. [2024]. A typical implementation uses a contrastive objective, often interpreted through an InfoNCE-based mutual-information (MI) surrogate, to encourage agreement between node-aligned views.

This global treatment works well on many graphs, yet it can become poorly calibrated. Topology encodes a relational inductive bias Chen and Kou [2023], whereas node attributes carry semantic information Li et al. [2025]; these two signals need not agree on the desired strength of cross-view alignment for every node Chen et al. [2025], Wan et al. [2024]. Anchors in structurally consistent regions may benefit from stronger alignment, whereas boundary, low-agreement (i.e., anchors whose two views disagree), or structurally noisy anchors may require weaker pressure. A single global objective cannot express this heterogeneity. The resulting calibration mismatch can over-align some

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

anchors while leaving others under-regularized, and can allow aggregate MI-surrogate estimates to drift during training Shou et al. [2025].

We therefore ask a focused question: can cross-view agreement in GCL be improved by treating the MI surrogate as a target-tracked control signal rather than as a quantity to be maximized uniformly? We study this question in contrastive node representation learning on attributed graphs, where cross-view alignment can be measured at the anchor level and evaluated through downstream node classification. This setting provides a controlled testbed for isolating how target-tracked MI-surrogate regulation changes GCL optimization, while still covering citation, co-purchase, academic, and lower-homophily graph structures.

MIRAGE addresses this calibration problem with a hierarchical regulation pipeline. The *core* mechanism contains two parts. First, anchor-wise MI setpoints assign bounded alignment budgets to individual nodes, allowing the encoder to apply stronger or weaker cross-view pressure based on each anchor's pre-regulation view embedding. Second, dual-view MI stabilization regularizes both the deviation of branch-level MI-surrogate estimates from a target and the dispersion of anchor-wise estimates. These two components constitute the core MI-regulation claim of this work.

MIRAGE also includes a *conditional* structural supplement. Some weak anchors may lack sufficient local context after the base dual-branch encoder. For these anchors, MIRAGE constructs higher-order aggregates from filtered multi-hop neighborhoods when the conditional hypergraph path is active. Local embedding similarity guides neighbor filtering and modulates the node-wise residual fusion weight, allowing the aggregates to supplement the original structural view. We therefore evaluate this path separately from the core MI-regulation mechanism, as a conditional compensation path for weak anchors.

Our empirical results follow this separation of concerns. Across six small-to-medium benchmarks, MIRAGE achieves competitive node-classification accuracy against strong GCL baselines, establishing a reliable performance basis for the mechanism analyses that follow. More importantly, mechanism-level analyses show that MIRAGE maintains MI-surrogate estimates near designated targets throughout training, whereas unregulated variants exhibit upward drift. Fixed-target sweeps further show that varying only the target level produces measurable changes in downstream performance. Together, these results support MIRAGE's central claim: anchor-wise MI allocation and view-level stabilization provide measurable control over GCL optimization in the evaluated node-classification setting. Our contributions are as follows:

- We formulate the GCL calibration problem through the lens of hierarchical MI-surrogate regulation, replacing uniform maximization with explicit anchor-wise and view-level control signals.
- We introduce MIRAGE, whose core consists of bounded anchor-wise MI setpoints and dual-view stabilization, complemented by a conditional hypergraph path that adaptively enhances structural representations for weak anchors.
- We provide a mechanism-centered empirical evaluation, including MI-surrogate trajectories, fixed-target sweeps, core-vs-conditional ablations, and component interaction studies, to show how target-tracked regulation changes GCL optimization beyond final node-classification accuracy.

2 Related Work

2.1 Graph Contrastive Learning

Current graph contrastive learning follows three main paradigms He et al. [2024a]. **InfoNCE-based Framework:** Maximizes mutual information (MI) to bring positive samples closer and push negative samples apart. ASP-adaptive extends this framework by integrating attributes and multi-scale views to address low MI Chen et al. [2025]. S3GCL combines fully connected MLP and cosine-parameterized spectral GNN for cross-channel contrast and introduces spatial neighbor augmentation for context-awareness, enabling homogeneity generalization and efficient inference Wan et al. [2024]. IFL-GCL reformulates GCL as PU learning, using the semantic alignment of InfoNCE to correct samples and eliminate negative sampling bias Wang et al. [2025]. **BGRL-like Framework:** Methods like BGRL use a dual-branch architecture with an online and target encoder to learn representations by

predicting alignments, without explicit negative samples Thakoor et al. [2021]. SGCL simplifies this further, showing that instance-level decorrelation is key to preventing model collapse Sun et al. [2024]. BLNN leverages graph homogeneity and a cross-attention mechanism to quantify positive support from neighbors, expanding positive samples while reducing noise and enhancing intra-class compactness Liu et al. [2024a]. **DGI-like Framework:** Represented by DGI, this approach treats all nodes as positive samples and generates a noisy distribution for negative samples, maximizing the JS divergence between them Veličković et al. [2018b]. GGD reveals that DGI focuses on group discrimination rather than mutual information maximization, simplifying it with BCE loss to distinguish positive and negative groups, which accelerates training Zheng et al. [2022]. PolyGCL builds on this by using learnable spectral polynomial filters to decouple high- and low-frequency views, enabling adaptive learning for graphs with varying compatibility Chen et al. [2024].

2.2 Graph Augmentation Strategies

Data augmentation is essential for constructing contrastive views. Common graph augmentation strategies include node dropping You et al. [2020], edge perturbation Zhang et al. [2024b], subgraph sampling Jiao et al. [2020], and attribute masking You et al. [2020]. Recent approaches, like GOUDA, unify these methods within a message-passing framework Zhuo et al. [2024].To overcome heuristic limitations, end-to-end learnable strategies have been proposed. GA2C models augmentation as a Markov decision process, using the Actor-Critic algorithm to learn continuous policies Liu et al. [2024b]. CI-GCL incorporates community structure invariance constraints to unify topology and feature augmentation objectives via spectral variation maximization Tan et al. [2024].Recent work also explores methods that remove the need for manual augmentation. For example, S3GCL uses cosine-parameterized Chebyshev polynomials to generate diverse spectral views, eliminating the reliance on data augmentation Wan et al. [2024]. GraphACL captures one-hop neighborhood information and two-hop monophily similarity through asymmetric views, adapting to both homophilic and heterophilic graphs without graph augmentation Xiao et al. [2023].

3 Methodology: MIRAGE

MIRAGE is a target-tracking regulation operator for node-aligned graph contrastive learning (GCL) views (Fig 1). Given a graph encoder that produces attribute-branch views and structure-branch views, MIRAGE regulates the InfoNCE-based MI surrogate at two coupled levels: anchor-wise allocation and branch-level stabilization. A conditional structural supplement is then applied only to weak anchors when structural signals appear reliable. This section focuses on the MIRAGE-specific regulation mechanism; the exact backbone instantiation, including GCN propagation, KNN graph construction, and the symmetrized contrastive denominator, is given in Appendix A.1.

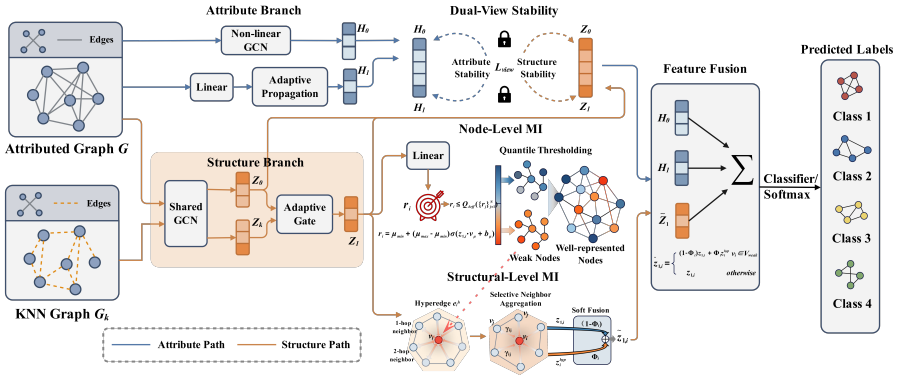


Figure 1: **Overview of MIRAGE.** MIRAGE regulates graph contrastive learning through anchor-wise MI setpoints and view-level stabilization, with a reliability-gated hypergraph supplement for weak anchors.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ be an attributed graph with N nodes, feature matrix $\mathbf{X} \in \mathbb{R}^{N \times d}$, adjacency matrix $\mathbf{A}$, and a feature-space KNN adjacency $\mathbf{A}_k$. The backbone encoder decouples attribute and structural representations Xu et al. [2023], producing two attribute views $(\mathbf{H}_0, \mathbf{H}_1)$ via adaptive

multi-scale propagation and two structure views $(\mathbf{Z}_0, \mathbf{Z}_1)$ via GCN encoders over the original and KNN-augmented graphs Chen et al. [2025]:

$$(\mathbf{H}_0, \mathbf{H}_1, \mathbf{Z}_0, \mathbf{Z}_1) = f_\theta(\mathbf{X}, \mathbf{A}, \mathbf{A}_k). \tag{1}$$

The base GCL objective aligns the two views within each branch:

$$\mathcal{L}_{\text{base}} = \lambda_1 \mathcal{C}(\mathbf{H}_0, \mathbf{H}_1) + \mathcal{C}(\mathbf{Z}_0, \mathbf{Z}_1), \tag{2}$$

where $\mathcal{C}(\cdot, \cdot)$ is a standard node-aligned contrastive loss.

Upon obtaining node-aligned views, a target-tracked MI regulation operator is jointly imposed across both branches via two control variables: an anchor-wise setpoint $\mu_i^{(b)}$ allocating alignment pressure per node, and a branch-level target μ^* stabilizing the MI surrogate trajectory during training. The conditional hypergraph enhancement pathway supplements weak anchors with higher-order structural signals, remaining architecturally independent of this core mechanism.

3.1 Anchor-Wise MI-Surrogate Allocation

A single global contrastive pressure cannot express that different anchors may require different degrees of cross-view alignment. For each branch $b \in \{\text{attr}, \text{struct}\}$, let $(\mathbf{U}_0^{(b)}, \mathbf{U}_1^{(b)})$ denote either $(\mathbf{H}_0, \mathbf{H}_1)$ or $(\mathbf{Z}_0, \mathbf{Z}_1)$, and let $\mathbf{u}_i^{0,b}, \mathbf{u}_i^{1,b}$ be the corresponding rows for anchor v_i. For a sampled node set $\mathcal{B}$, we use the anchor-wise cross-view InfoNCE loss Oord et al. [2018]

$$\ell_i^{(b)} = -\log \frac{e^{(\text{sim}(\mathbf{u}_i^{0,b}, \mathbf{u}_i^{1,b})/\tau)}}{\sum_{j \in \mathcal{B}} e^{(\text{sim}(\mathbf{u}_i^{0,b}, \mathbf{u}_j^{1,b})/\tau)}}, \tag{3}$$

and define the tracked MI surrogate as

$$\widehat{I}_i^{(b)} = \log |\mathcal{B}| - \ell_i^{(b)}. \tag{4}$$

For full-graph training, $|\mathcal{B}| = N$; for sampled training, it is the negative-pool size. We emphasize that $\widehat{I}_i^{(b)}$ is an InfoNCE-based surrogate attached to anchor i, not the mutual information of an isolated node.

MIRAGE assigns each anchor a bounded setpoint for this surrogate. The setpoint is generated from the pre-regulation view embedding through a lightweight setpoint head:

$$\mu_i^{(b)} = \mu_{\min} + (\mu_{\max} - \mu_{\min}) \cdot \sigma\left(\mathbf{p}_i^{(0,b)} \mathbf{v}_\mu + b_\mu\right), \tag{5}$$

Here $p_i^{(0,\text{attr})} = h_i^0$ and $p_i^{(0,\text{struct})} = z_i^0$. The scaled sigmoid bounds the setpoints in $[\mu_{\min}, \mu_{\max}]$. The setpoint head and its input encoder are trained jointly.

The anchor-wise allocation loss is

$$\mathcal{L}_{\text{alloc}} = \frac{1}{N} \sum_{i=1}^{N} w_i \sum_{b \in \{\text{attr}, \text{struct}\}} \left(\widehat{I}_i^{(b)} - \mu_i^{(b)}\right)^2, \tag{6}$$

where $w_i = d_i / \max_j d_j$ is a normalized degree weight. The degree weight affects only the importance of the anchor in the loss; it does not change the setpoint itself. Under Eq. (5), the allocation loss updates the encoder through both the MI-surrogate branch $\hat{I}_i^{(b)}$ and the setpoint branch $\mu_i^{(b)}$, while also updating the setpoint-head parameters through $\mu_i^{(b)}$. Thus, $\mu_i^{(b)}$ is a bounded, trainable allocation rule for alignment pressure, rather than an estimator of a ground-truth optimal MI value.

The allocation loss in Eq. (6) changes the equilibrium of the anchor-wise surrogate rather than only changing the magnitude of an InfoNCE gradient. A per-anchor weight would still optimize every selected anchor in the same monotone direction, namely toward stronger positive-to-negative separation. By contrast, MIRAGE defines an anchor-specific operating point. If $\widehat{I}_i^{(b)} < \mu_i^{(b)}$, the regulation term increases the effective alignment pressure; if $\widehat{I}_i^{(b)} > \mu_i^{(b)}$, it discourages further surrogate growth. The setpoint therefore acts as a bidirectional control signal, not as a scalar importance weight.

Numerically, matching $\widehat{I}_i^{(b)}$ to a bounded target keeps the anchor's negative log-softmax loss in a non-trivial range, rather than forcing uniform maximization for all anchors. The non-triviality of this allocation is tested empirically through the fixed-target sweep in Sec. 4.4 (Fig 4), which shows that varying the target itself moves downstream accuracy.

In the node-adaptive configuration, we precompute a feature-based neighbor reference for each node. Let $\mathcal{N}(i) = \{j : (i,j) \in E\}$ and let x_i denote the input feature vector. We set $s_i = |\mathcal{N}(i)|^{-1} \sum_{j \in \mathcal{N}(i)} \cos(x_i, x_j)$, with $s_i = 0$ for an isolated node. We then normalize the values across nodes as $\widetilde{s}_i = (s_i - \min_v s_v)/(\max_v s_v - \min_v s_v)$ and define $m_i^{\text{neigh}} = \mu_{\min} + (\mu_{\max} - \mu_{\min})\widetilde{s}_i$. If all s_i are equal, we set $\widetilde{s}_i = 0.5$. The corresponding deviation term is

$$\mathcal{L}_{\text{ideal}} = \frac{\lambda_{\text{ideal}}}{N} \sum_{i=1}^{N} \sum_{b \in \{\text{attr,struct}\}} \big(\hat{I}_i^{(b)} - m_i^{\text{neigh}}\big)^2. \tag{7}$$

3.2 Branch-Level MI-Surrogate Stabilization

Anchor-wise allocation controls where alignment pressure is placed. We also regularize how the surrogate evolves at the branch level. For branch b, let

$$\widehat{I}^{(b)} = \frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \widehat{I}_i^{(b)}. \tag{8}$$

The view-level stabilization loss is

$$\mathcal{L}_{\text{view}} = \sum_{b \in \{\text{attr,struct}\}} \left[\lambda_{\text{dev}} \left(\widehat{I}^{(b)} - \mu^* \right)^2 + \lambda_{\text{var}} \operatorname{Var}\Big(\{\widehat{I}_i^{(b)}\}_{i \in \mathcal{B}} \Big) \right]. \tag{9}$$

Here μ^* is an independent branch-level target selected by validation; it is not the average of the anchor-wise setpoints. Eq. (6) controls anchor-level allocation, whereas Eq. (9) controls branch-level drift and high dispersion across anchors. The variance term does not suppress per-anchor diversity; it penalizes runaway dispersion of the realized surrogate, while the node-specific variation in $\mu_i^{(b)}$ is preserved by the bounded setpoint form in Eq. (5) and the allocation loss in Eq. (6). We refer to Eq. (9) as *view-level stabilization* because it penalizes both systematic deviation from μ^* and excessive across-anchor spread; Sec. 4.4 (Fig. 3) verifies target tracking against an unregulated baseline.

The resulting core regulation term is

$$\mathcal{L}_{\text{reg}} = \lambda_{\text{MI}}(\mathcal{L}_{\text{alloc}} + \mathcal{L}_{\text{ideal}}) + \mathcal{L}_{\text{view}} \tag{10}$$

3.3 Conditional Structural Supplement

The structural supplement is not the main MI-regulation claim. It is a conditional path for weak anchors, activated only after warmup and regulated by global structural reliability and local neighbor filtering.

We select weak anchors using a separate score obtained by applying the shared setpoint head, without gradient, to the pre-supplement structural view $\mathbf{Z}_1$ of Eq. (19). Let $\mathbf{z}_{1,i}$ denote row i of $\mathbf{Z}_1$. We define

$$r_i = \mu_{\min} + (\mu_{\max} - \mu_{\min})\, \sigma(\mathbf{z}_{1,i}\mathbf{v}_\mu + b_\mu), \qquad \mathcal{V}_{\text{weak}} = \big\{ v_i : \; r_i \le Q_{\delta_{\text{eff}}}\big(\{r_j\}_{j=1}^N\big) \big\}. \tag{11}$$

Here $Q_{\delta_{\text{eff}}}$ denotes the δ_{eff}-quantile, with $\delta_{\text{eff}} = \alpha\delta_{\text{base}}$ when the structural supplement is enabled, as defined in Appendix A.2. The score r_i is used only for weak-anchor selection, whereas $\mu_i^{(\text{struct})}$ in Eqs. (5)–(6) is computed from $\mathbf{z}_{0,i}$ for the allocation loss. A low r_i is a model-derived selection signal, not an oracle difficulty label.

For each weak anchor, a higher-order representation $\mathbf{z}_i^{\text{hyp}}$ is computed from filtered k_h-hop neighbors using the procedure in Appendix A.2. MIRAGE then uses a residual fusion gate:

$$\widetilde{\mathbf{z}}_{1,i} = (1 - \phi_i)\mathbf{z}_{1,i} + \phi_i \mathbf{z}_i^{\text{hyp}}, \tag{12}$$

where ϕ_i denotes the residual fusion weight for node i, which is modulated by a graph-level scaling factor and local embedding similarity. The conditional hypergraph path provides structural compensation for weak anchors through neighbor filtering and residual fusion. Further implementation details are provided in Appendix A.2.

Let $\widehat{I}_i^{\mathrm{sup}}$ denote the per-anchor surrogate computed between $\mathbf{z}_i^0$ and $\widetilde{\mathbf{z}}_{1,i}$ by the same formula as Eq. (4), and let $\widehat{I}(\mathbf{Z}_0;\widetilde{\mathbf{Z}}_1) = \frac{1}{N}\sum_{i=1}^{N}\widehat{I}_i^{\mathrm{sup}}$ be its branch-level aggregate. The small offset $\epsilon \geq 0$ lets the supplement target sit slightly above μ^*, and η is kept at the same grouped auxiliary scale as λ_{var}. The optional supplement loss is

$$\mathcal{L}_{\mathrm{sup}} = \lambda_{\mathrm{hyper}}\left[\left(\widehat{I}(\mathbf{Z}_0;\widetilde{\mathbf{Z}}_1) - (\mu^* + \epsilon)\right)^2 + \eta\,\mathrm{Var}\left(\{\widehat{I}_i^{\mathrm{sup}}\}_{i=1}^{N}\right)\right], \tag{13}$$

where $\widetilde{\mathbf{Z}}_1$ equals $\mathbf{Z}_1$ for non-weak anchors and the fused representation in Eq. (12) for weak anchors. The term is applied only after the warmup epoch E_w.

3.4 Training Objective and Output Embedding

MIRAGE can be summarized as a base contrastive learner plus a two-level regulation term and an optional delayed supplement:

$$\mathcal{L} = \mathcal{L}_{\mathrm{base}} + \mathcal{L}_{\mathrm{reg}} + \mathbb{I}[t \geq E_w]\mathcal{L}_{\mathrm{sup}}. \tag{14}$$

When structural fusion is active, the fused structural view is used in the structure-dependent terms of both $\mathcal{L}_{\mathrm{base}}$ and $\mathcal{L}_{\mathrm{reg}}$, $\mathcal{L}_{\mathrm{sup}}$ is an additional auxiliary term.

This grouping is important: $\mathcal{L}_{\mathrm{alloc}}$ and $\mathcal{L}_{\mathrm{view}}$ are the two levels of the same MI-surrogate regulation mechanism, while $\mathcal{L}_{\mathrm{sup}}$ is delayed and conditional. In practice, λ_{MI} is the main tuning knob (Eqs. (10)); the view-level and supplement weights are kept small and varied together in the grouped sensitivity analysis.

For downstream evaluation, we use the fused embedding

$$\mathbf{E} = \mathbf{H}_0 + \mathbf{H}_1 + \widetilde{\mathbf{Z}}_1. \tag{15}$$

4 Experiments

We evaluate MIRAGE as a self-supervised node representation method. The experiments are organized around four questions, each tied to a specific claim of the method rather than to downstream accuracy alone:

- **RQ1 (competitiveness).** On standard node-classification benchmarks, does MIRAGE remain competitive with representative unsupervised, semi-supervised, and GCL baselines?
- **RQ2 (component complementarity).** When individual MIRAGE components are removed, and when the MI-regulation core is separated from the conditional structural supplement, do the remaining components still show coherent behavior?
- **RQ3 (target-tracking mechanism).** Does MIRAGE actually track the anchor-wise MI-surrogate setpoints during training, and does moving the target itself change downstream accuracy?
- **RQ4 (tuning and safeguards).** Among MIRAGE's loss weights, which variables act as primary knobs, and does the structure-reliability gate behave as a safeguard rather than as an additional performance-tuned component?

Dataset statistics, implementation details, and extended diagnostics are given in Appendix D–E. Appendix E provides extended diagnostics, including parameter sensitivity, hypergraph warmup sweeps, perturbation stability, progressive ablations, gate diagnostics, and qualitative visualization.

4.1 Datasets and Experimental Settings

Datasets. We evaluate MIRAGE on the six benchmarks: Cora Sen et al. [2008], Citeseer Sen et al. [2008], WikiCS Mernyei and Cangea [2020], Amazon-Photo McAuley et al. [2015], ACM Wang et al.

Table 1: Node classification accuracy (%) on six benchmarks. Best results are in **bold** and second best are underlined.

Method	Cora	Citeseer	WikiCS	Amazon-Photo	ACM	Cora-ML
Node2vec	76.52 ± 0.94	52.11 ± 1.71	77.15 ± 0.33	89.49 ± 0.44	68.87 ± 1.12	78.68 ± 1.12
DeepWalk	74.96 ± 2.16	50.06 ± 1.30	77.36 ± 0.67	90.30 ± 0.25	70.31 ± 0.96	77.80 ± 2.39
GCN	79.88 ± 0.78	69.62 ± 0.60	79.41 ± 0.50	91.65 ± 0.47	89.93 ± 0.75	85.28 ± 0.64
GAT	83.00 ± 0.70	72.50 ± 0.70	**80.97 ± 0.38**	92.13 ± 0.38	90.70 ± 0.62	**86.08 ± 0.69**
DGI	82.30 ± 0.60	71.80 ± 0.70	61.01 ± 1.38	93.29 ± 0.34	91.98 ± 0.51	85.66 ± 0.80
GRACE	83.30 ± 0.40	72.10 ± 0.50	78.84 ± 0.24	89.25 ± 0.51	90.12 ± 0.85	83.05 ± 0.91
BGRL	75.92 ± 1.04	60.01 ± 1.72	79.36 ± 0.53	92.87 ± 0.27	84.07 ± 0.96	77.18 ± 0.86
AFGRL	75.12 ± 1.33	63.50 ± 1.33	77.62 ± 0.49	93.22 ± 0.28	87.28 ± 0.69	55.66 ± 3.63
SP-GCL	83.16 ± 0.13	71.96 ± 0.42	79.01 ± 0.51	92.49 ± 0.31	83.08 ± 0.96	81.78 ± 1.11
HomoGCL	84.50 ± 0.50	72.30 ± 0.70	80.16 ± 0.45	92.92 ± 0.18	85.85 ± 3.11	79.71 ± 1.56
NCLA	82.20 ± 1.60	71.70 ± 0.90	79.09 ± 0.48	90.20 ± 1.30	91.85 ± 0.37	82.29 ± 0.90
SPIKEGCL	80.02 ± 0.85	66.66 ± 1.12	79.19 ± 0.62	93.00 ± 0.10	87.41 ± 0.59	78.15 ± 2.62
PiGCL	84.63 ± 0.78	73.51 ± 0.64	78.82 ± 0.50	93.14 ± 0.30	90.90 ± 0.64	83.33 ± 1.01
iGCL	81.60 ± 1.00	70.20 ± 0.90	78.22 ± 0.69	93.10 ± 0.26	79.73 ± 1.24	81.08 ± 1.42
ML^2-GCL	83.00 ± 1.30	71.70 ± 1.00	73.30 ± 1.70	90.70 ± 1.60	84.46 ± 0.96	80.33 ± 1.20
GTCA	82.50 ± 1.30	69.70 ± 1.70	69.70 ± 1.50	90.50 ± 1.20	78.94 ± 4.67	78.38 ± 1.34
MIRAGE	**85.21 ± 0.24**	**73.94 ± 0.73**	80.62 ± 0.27	**93.41 ± 0.21**	**92.57 ± 0.31**	85.96 ± 0.52

[2019], and Cora-ML Bojchevski and Günnemann [2017]. To further examine MIRAGE under a lower-homophily setting, we additionally include Minesweeper Platonov et al. [2023] for a diagnostic evaluation, with results reported in Appendix D.

Baselines. We compare with unsupervised graph learning methods (DeepWalk Perozzi et al. [2014], Node2Vec Grover and Leskovec [2016]), semi-supervised GNNs (GCN Jiang et al. [2019], GAT Veličković et al. [2018a]), and self-supervised GCL methods including DGI Veličković et al. [2018b], GRACE Zhu et al. [2020], BGRL Thakoor et al. [2021], AFGRL Lee et al. [2022], SP-GCL Wang et al. [2022], HomoGCL Li et al. [2023c], NCLA Shen et al. [2023], SPIKEGCL Li et al. [2023b], PiGCL He et al. [2024b], iGCL Li et al. [2023a], ML^2-GCL Liang et al. [2025a], and GTCA Liang et al. [2025b]. For baseline results, we follow official implementations and standard evaluation protocols whenever available, supplemented by authors' reported results for missing entries.

4.2 Competitive Node Classification (RQ1)

Table 1 reports node-classification accuracy on six small-to-medium attributed graph benchmarks. MIRAGE achieves the best overall result on four datasets: Cora, Citeseer, Amazon-Photo, and ACM. These wins cover citation networks, product co-purchase graphs, and academic information networks, suggesting that MI-surrogate regulation provides a competitive training principle across different graph domains. Compared with representative graph embedding, GNN, and graph contrastive baselines, MIRAGE consistently ranks among the strongest methods in the benchmark suite. The largest gains appear on ACM and Citeseer, where regulating anchor-wise and branch-level MI-surrogate behavior improves the balance between cross-view agreement and representation stability. On WikiCS and Cora-ML, MIRAGE remains competitive with the strongest supervised or specialized baselines, indicating that the regulation mechanism does not trade off representation quality for controllability.

4.3 Component Complementarity (RQ2)

Fig. 2 performs a single-removal ablation of the three MIRAGE components. The drops are not uniform, which is expected because the modules operate at different levels of the hierarchy. The setpoint module has the clearest effect on Cora and Citeseer, supporting the importance of anchor-wise allocation for calibrating contrastive pressure. The reliability gate is most consequential on WikiCS, where gating out unreliable structural supplementation is more effective than applying the hypergraph path indiscriminately. View-level stabilization produces smaller final-accuracy changes, but its role is complementary: it constrains the branch-level surrogate trajectory, whose effect is examined more directly in Sec. 4.4 and Appendix E.5. Together, these results support a hierarchical interpretation of

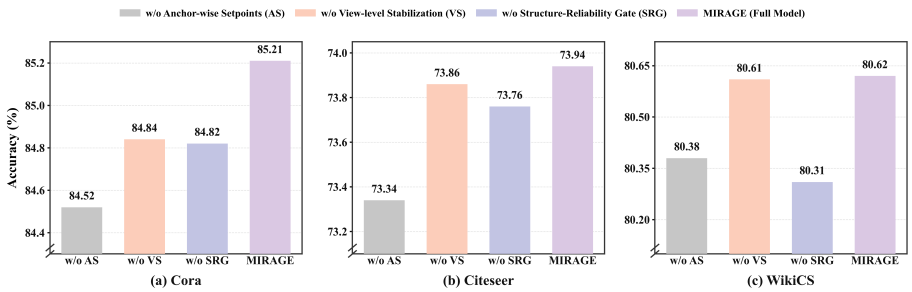


Figure 2: Ablation study of MIRAGE on three benchmark datasets. Each variant removes one component: anchor-wise setpoints, view-level stabilization, or the structure-reliability gate.

Table 2: Core-only ablation separating MI regulation from the structural supplement.

Variant	Cora	ACM
MIRAGE core only (no structural supplement)	84.95 ± 0.21	91.27 ± 0.08
Full MIRAGE	$\mathbf{85.21 \pm 0.24}$	$\mathbf{92.57 \pm 0.31}$

MIRAGE: the setpoints allocate anchor-level pressure, the stabilization term controls view-level drift, and the gate decides when structural compensation should be used.

To separate the core MI-regulation operator from the conditional structural supplement, Table 2 disables the hypergraph path. On Cora, the core-only model remains within 0.3% of the full model, indicating that anchor-wise setpoint allocation and view-level target tracking already form a strong regulated GCL learner. On ACM, the supplement contributes more substantially, which is consistent with reliable higher-order neighborhoods providing useful compensation for weak anchors. This pattern supports the intended hierarchy of MIRAGE: the hypergraph path is not a replacement for the MI-regulation core, but a conditional extension that becomes useful when local structural signals are reliable. A progressive ablation of the combined node-level terms $\mathcal{L}_{\text{alloc}} + \mathcal{L}_{\text{ideal}}$ and the view-level term $\mathcal{L}_{\text{view}}$ is given in Appendix E.7.

4.4 MI Target-Tracking Mechanism (RQ3)

RQ3 examines whether MIRAGE controls the intended optimization variable: the InfoNCE-based MI surrogate. Fig. 3 plots the mean anchor-wise MI surrogate during training on ACM and Citeseer. The unregulated InfoNCE variant continues to increase and eventually drifts to approximately $[1.85, 1.90]$, whereas MIRAGE keeps the surrogate close to the prescribed target level on both datasets. This trajectory-level diagnostic verifies that the regulation term acts at the surrogate-dynamics level: it changes the training path followed by the contrastive objective, rather than being assessed only through the final linear-evaluation score.

We next test whether the target value itself has behavioral consequences under a fixed architecture and training pipeline. Fig. 4 sweeps a single fixed MI-surrogate setpoint $\mu \in \{0.3, 0.5, 0.7, 0.9, 1.1, 1.3\}$ and compares it with the adaptive setpoint interval $(0.6, 0.8)$. On Citeseer, changing only the target value produces a clear target-response pattern: accuracy varies by 2.31 percentage points across fixed targets, and overly high targets degrade performance. This shows that the MI-surrogate target functions as an operational control variable: under the same backbone and non-target settings, different target values lead to different learned representations. On Cora, the response curve is flatter, suggesting that a strongly homophilous graph can be less sensitive to the precise target once the surrogate lies in a reasonable operating range. The adaptive interval remains in the best-performing region on both datasets, supporting bounded target-tracked calibration rather than uniform surrogate maximization.

4.5 Tuning and Safeguards (RQ4)

MIRAGE contains several coefficients, so we explicitly test whether it behaves like many independent knobs. Table 3 groups the auxiliary stability and supplement weights and scales them jointly by $\{0.5\times, 1.0\times, 2.0\times\}$ under three values of λ_{MI} on Cora. For any fixed λ_{MI}, changing the auxiliary scale alters accuracy by less than 0.5 percentage points; at a fixed auxiliary scale, changing λ_{MI} alters

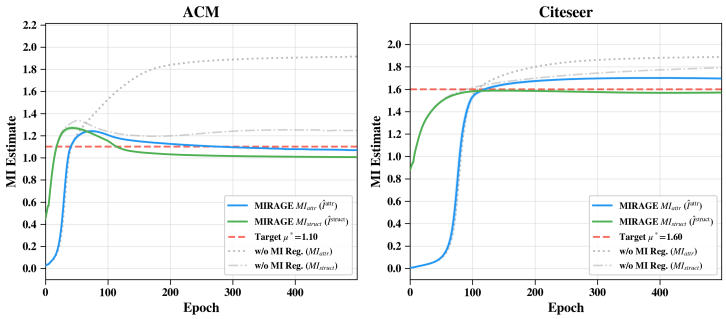


Figure 3: Epoch-wise MI-surrogate trajectories during training on ACM and Citeseer. MIRAGE keeps the surrogate close to the prescribed target level (red dashed line), whereas the unregulated variant drifts upward.

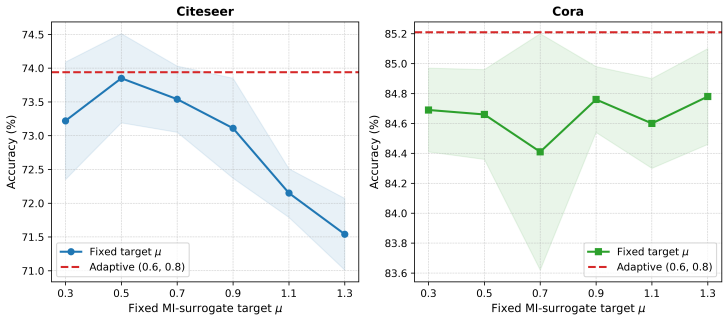


Figure 4: Target-response curves under fixed and adaptive MI-surrogate targets. Solid curves show mean accuracy for fixed target values, shaded bands denote one standard deviation over seeds, and red dashed lines indicate the mean accuracy obtained with the adaptive setpoint interval $(0.6, 0.8)$.

accuracy by up to 4.07 percentage points. This supports a simple tuning recipe: select λ_{MI} first, then treat the view-level and supplement weights as a grouped auxiliary scale. Additional one-dimensional sensitivity analyses for λ_{MI}, the residual fusion weight ϕ_{base}, and the warmup schedule are kept in Appendix E.1–E.4.

Table 3: Grouped sensitivity analysis on Cora evaluating loss-term interactions (%). The auxiliary scale applies jointly to λ_{dev}, λ_{var}, and λ_{hyper}.

Aux. Scale \ λ_{MI}	0.1	0.3	0.5
0.5×	84.59 ± 1.00	83.73 ± 0.77	80.52 ± 1.91
1.0×	84.11 ± 1.48	83.65 ± 1.01	80.68 ± 1.31
2.0×	84.24 ± 1.36	83.70 ± 0.76	80.82 ± 1.29

The structure-reliability gate is evaluated in Appendix E.6. The gated variant outperforms always-on aggregation on Cora and ACM and matches the core-only model on Citeseer, where the structural supplement would otherwise be harmful. These results support the intended role of the gate as a conservative safeguard: it enables higher-order aggregation when the graph structure is reliable and local neighbors are filtered, while suppressing the supplement when indiscriminate aggregation is likely to inject noise. Thus, the gate complements the MI-regulation core by controlling when the conditional structural path should be activated, rather than functioning as an independent performance-tuned branch.

5 Limitations and Conclusion

MIRAGE presents target-tracked MI-surrogate regulation as a practical mechanism for calibrating cross-view agreement in node-aligned graph contrastive learning. The current evaluation focuses on self-supervised node representation learning with linear probing, where anchor-wise surrogate

estimates, branch-level trajectories, and downstream accuracy can be examined jointly. Extending this regulation principle to link prediction, graph classification, few-shot evaluation, and large-scale graph pretraining remains an important direction.

Several limitations remain. Full-graph MI-surrogate tracking relies on pairwise similarities and is therefore quadratic in the number of nodes unless negative sampling or approximate similarity computation is used. Our experiments therefore focus on small-to-medium graphs, and scalability beyond 100K nodes has not been validated. The benchmark suite is mostly moderate-to-high homophily, with Minesweeper serving as a lower-homophily stress case; thus, the structure-reliability gate should be interpreted as a safeguard against harmful higher-order aggregation, not as a dedicated solution for strongly heterophilous graphs. MIRAGE also introduces more hyperparameters than single-loss GCL baselines, although the grouped sensitivity analysis provides a structured tuning recipe.

Overall, MIRAGE shows that the InfoNCE-based MI surrogate need not be treated only as a scalar objective to maximize. By assigning bounded anchor-wise setpoints and stabilizing branch-level surrogate trajectories, MIRAGE turns cross-view agreement into an explicit regulation signal. The empirical results show competitive node representations, while the mechanism analyses demonstrate target tracking, target sensitivity, and the complementary roles of anchor-level allocation, view-level stabilization, and conditional structural supplementation. These findings support MI-surrogate regulation as a useful design principle for graph contrastive learning and motivate its extension to larger graphs and broader graph learning tasks.

6 Acknowledgments and Disclosure of Funding

This work was supported by the Special Research Project for Building a Strong Education Province in Henan, China, entitled "A Study on the Classification-Based Evaluation of the Contributions of Philosophy and Social Science Research Outcomes to Their Translation and Application at Henan Universities and on Data Governance Mechanisms."

References

Aleksandar Bojchevski and Stephan Günnemann. Deep gaussian embedding of graphs: Unsupervised inductive learning via ranking. *arXiv preprint arXiv:1707.03815*, 2017.

Jialu Chen and Gang Kou. Attribute and structure preserving graph contrastive learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 7024–7032, 2023.

Jialu Chen, Rui Chen, and Gang Kou. Unifying attribute and structure preservation for enhanced graph contrastive learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(6): 10479–10490, 2025.

Jingyu Chen, Runlin Lei, and Zhewei Wei. Polygcl: Graph contrastive learning via learnable spectral polynomial filters. In *The twelfth international conference on learning representations*, 2024.

Shikun Feng, Yuyan Ni, Minghao Li, Yanwen Huang, Zhi-Ming Ma, Wei-Ying Ma, and Yanyan Lan. Unicorn: A unified contrastive learning approach for multi-view molecular representation learning. In *International Conference on Machine Learning*, pages 13256–13277. PMLR, 2024.

Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

Dongxiao He, Lianze Shan, Jitao Zhao, Hengrui Zhang, Zhen Wang, and Weixiong Zhang. Exploitation of a latent mechanism in graph contrastive learning: Representation scattering. *Advances in Neural Information Processing Systems*, 37:115351–115376, 2024a.

Dongxiao He, Jitao Zhao, Cuiying Huo, Yongqi Huang, Yuxiao Huang, and Zhiyong Feng. A new mechanism for eliminating implicit conflict in graph contrastive learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 12340–12348, 2024b.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Bo Jiang, Ziyan Zhang, Doudou Lin, Jin Tang, and Bin Luo. Semi-supervised learning with graph learning-convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11313–11320, 2019.

Yizhu Jiao, Yun Xiong, Jiawei Zhang, Yao Zhang, Tianqi Zhang, and Yangyong Zhu. Sub-graph contrast for scalable self-supervised graph representation learning. In *2020 IEEE international conference on data mining (ICDM)*, pages 222–231. IEEE, 2020.

Di Jin, Zhizhi Yu, Cuiying Huo, Rui Wang, Xiao Wang, Dongxiao He, and Jiawei Han. Universal graph convolutional networks. *Advances in neural information processing systems*, 34:10654–10664, 2021.

Namkyeong Lee, Junseok Lee, and Chanyoung Park. Augmentation-free self-supervised learning on graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 7372–7380, 2022.

Haifeng Li, Jun Cao, Jiawei Zhu, Qinyao Luo, Silu He, and Xuying Wang. Augmentation-free graph contrastive learning of invariant-discriminative representations. *IEEE Transactions on Neural Networks and Learning Systems*, 35(8):11157–11167, 2023a.

Jintang Li, Huizhe Zhang, Ruofan Wu, Zulun Zhu, Baokun Wang, Changhua Meng, Zibin Zheng, and Liang Chen. A graph is worth 1-bit spikes: When graph contrastive learning meets spiking neural networks. *arXiv preprint arXiv:2305.19306*, 2023b.

Wen Li, Kai Yang, and Kairong Li. Structure-augmentation and attribute-aware graph contrastive learning with weak information. *Knowledge-Based Systems*, page 114015, 2025.

Wen-Zhi Li, Chang-Dong Wang, Hui Xiong, and Jian-Huang Lai. Homogcl: Rethinking homophily in graph contrastive learning. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 1341–1352, 2023c.

Jianqing Liang, Zhiqiang Li, Xinkai Wei, Yuan Liu, and Zhiqiang Wang. MI2-gcl: Manifold learning inspired lightweight graph contrastive learning. In *Forty-second International Conference on Machine Learning*, 2025a.

Jianqing Liang, Xinkai Wei, Min Chen, Zhiqiang Wang, and Jiye Liang. Gnn-transformer cooperative architecture for trustworthy graph contrastive learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 18667–18675, 2025b.

Yunhui Liu, Huaisong Zhang, Tieke He, Tao Zheng, and Jianhua Zhao. Bootstrap latents of nodes and neighbors for graph self-supervised learning. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 76–92. Springer, 2024a.

Ziyang Liu, Chaokun Wang, and Cheng Wu. Graph contrastive learning with reinforcement augmentation. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 2225–2233, 2024b.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52, 2015.

Hanyang Meng, Jielong Yang, and Li Peng. Mitigating over-smoothing in graph neural networks via separation coefficient guided adaptive graph structure adjustment. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pages 5959–5966, 2025.

Péter Mernyei and Cătălina Cangea. Wiki-cs: A wikipedia-based benchmark for graph neural networks. *arXiv preprint arXiv:2007.02901*, 2020.

Xuelian Ni, Fei Xiong, Yu Zheng, and Liang Wang. Graph contrastive learning with kernel dependence maximization for social recommendation. In *Proceedings of the ACM Web Conference 2024*, pages 481–492, 2024.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of gnns under heterophily: Are we really making progress? *arXiv preprint arXiv:2302.11640*, 2023.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

Xiao Shen, Dewang Sun, Shirui Pan, Xi Zhou, and Laurence T Yang. Neighbor contrastive learning on learnable graph augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 9782–9791, 2023.

Yuntao Shou, Haozhi Lan, and Xiangyong Cao. Contrastive graph representation learning with adversarial cross-view reconstruction and information bottleneck. *Neural Networks*, 184:107094, 2025.

Wangbin Sun, Jintang Li, Liang Chen, Bingzhe Wu, Yatao Bian, and Zibin Zheng. Rethinking and simplifying bootstrapped graph latents. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 665–673, 2024.

Shiyin Tan, Dongyuan Li, Renhe Jiang, Ying Zhang, and Manabu Okumura. Community-invariant graph contrastive learning. In *International Conference on Machine Learning*, pages 47579–47606. PMLR, 2024.

Shantanu Thakoor, Corentin Tallec, Mohammad Gheshlaghi Azar, Rémi Munos, Petar Veličković, and Michal Valko. Bootstrapped representation learning on graphs. In *ICLR 2021 workshop on geometrical and topological representation learning*, pages 1–14. OpenReview. net, 2021.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. Graph attention networks. In *International conference on learning representations*, volume 6. Ithaca, 2018a.

Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018b.

Guancheng Wan, Yijun Tian, Wenke Huang, Nitesh V Chawla, and Mang Ye. S3gcl: Spectral, swift, spatial graph contrastive learning. In *Forty-first International Conference on Machine Learning*, 2024.

Haonan Wang, Jieyu Zhang, Qi Zhu, Wei Huang, Kenji Kawaguchi, and Xiaokui Xiao. Single-pass contrastive learning can work for both homophilic and heterophilic graph. *arXiv preprint arXiv:2211.10890*, 2022.

Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *The world wide web conference*, pages 2022–2032, 2019.

Zixu Wang, Bingbing Xu, Yige Yuan, Huawei Shen, and Xueqi Cheng. Infonce is a free lunch for semantically guided graph contrastive learning. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 719–728, 2025.

Teng Xiao, Huaisheng Zhu, Zhengyu Chen, and Suhang Wang. Simple and asymmetric graph contrastive learning without augmentations. *Advances in neural information processing systems*, 36:16129–16152, 2023.

Fei Xiong, Tao Zhang, Shirui Pan, Guixun Luo, and Liang Wang. Robust graph based social recommendation through contrastive multi-view learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12890–12898, 2025.

Hongcai Xu, Junpeng Bao, and Wenbo Liu. Double-branch multi-attention based graph neural network for knowledge graph completion. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 15257–15271, 2023.

Yuhua Xu, Junli Wang, Mingjian Guang, Chungang Yan, and Changjun Jiang. Graph contrastive learning with min-max mutual information. *Information Sciences*, 665:120378, 2024.

Liang Yang, Zhenna Li, Jiaming Zhuo, Jing Liu, Ziyi Ma, Chuan Wang, Zhen Wang, and Xiaochun Cao. Graph contrastive learning with joint spectral augmentation of attribute and topology. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 21983–21991, 2025.

Shaochen Yang, Zhaorun Liao, Runyu Chen, Yuren Lai, and Wei Xu. Multi-view fair-augmentation contrastive graph clustering with reliable pseudo-labels. *Information Sciences*, 674:120739, 2024a.

Yonghui Yang, Le Wu, Zihan Wang, Zhuangzhuang He, Richang Hong, and Meng Wang. Graph bottlenecked social recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3853–3862, 2024b.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33: 5812–5823, 2020.

Shengzhe Zhang, Liyi Chen, Chao Wang, Shuangli Li, and Hui Xiong. Temporal graph contrastive learning for sequential recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 9359–9367, 2024a.

Zeyu Zhang, Lu Li, Shuyan Wan, Sijie Wang, Zhiyi Wang, Zhiyuan Lu, Dong Hao, and Wanli Li. Dropedge not foolproof: effective augmentation method for signed graph neural networks. *Advances in Neural Information Processing Systems*, 37:117041–117069, 2024b.

Yizhen Zheng, Shirui Pan, Vincent Lee, Yu Zheng, and Philip S Yu. Rethinking and scaling up graph contrastive learning: An extremely efficient approach with group discrimination. *Advances in Neural Information Processing Systems*, 35:10809–10820, 2022.

Gengmo Zhou, Zhen Wang, Feng Yu, Guolin Ke, Zhewei Wei, and Zhifeng Gao. S-molsearch: 3d semi-supervised contrastive learning for bioactive molecule search. *Advances in Neural Information Processing Systems*, 37:74715–74737, 2024.

Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep graph contrastive representation learning. *arXiv preprint arXiv:2006.04131*, 2020.

Jiaming Zhuo, Yintong Lu, Hui Ning, Kun Fu, Bingxin Niu, Dongxiao He, Chuan Wang, Yuanfang Guo, Zhen Wang, Xiaochun Cao, et al. Unified graph augmentations for generalized contrastive learning on graphs. *Advances in Neural Information Processing Systems*, 37:37473–37503, 2024.

A Model and Training Details

A.1 Backbone View Instantiation

This appendix gives the concrete encoder used to instantiate Eq. (1). The main paper treats this encoder as a standard node-aligned GCL backbone and focuses on MIRAGE's target-tracked regulation operator.

Let $\widehat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ and $\widetilde{\mathbf{A}} = \widehat{\mathbf{D}}^{-1/2}\widehat{\mathbf{A}}\widehat{\mathbf{D}}^{-1/2}$. The first attribute view is produced by a two-layer GCN:

$$\mathbf{H}_0 = \widetilde{\mathbf{A}}\,\mathrm{ReLU}\Big(\widetilde{\mathbf{A}}\mathbf{X}\mathbf{\Theta}^{(0)}\Big)\,\mathbf{\Theta}^{(1)}. \tag{16}$$

The second attribute view uses adaptive propagation Meng et al. [2025]. With $\mathbf{P}^{(0)} = \mathbf{X}\mathbf{\Theta}_{\mathrm{lin}}$ and $\mathbf{P}^{(r)} = \widetilde{\mathbf{A}}\mathbf{P}^{(r-1)}$, we define

$$\mathbf{H}_1 = \sum_{r=0}^{K} \boldsymbol{\alpha}^{(r)}\mathbf{P}^{(r)}, \qquad \boldsymbol{\alpha}^{(r)} = \sigma(\mathbf{P}^{(r)}\mathbf{v}_a + b_a). \tag{17}$$

For the structure branch, a shared GCN encoder f_s produces a topology view from the original graph and an auxiliary feature-geometric view from the KNN graph:

$$\mathbf{Z}_0 = f_s^{(2)}(\mathbf{X}, \mathbf{A}), \qquad \mathbf{Z}_k = f_s^{(1)}(\mathbf{X}, \mathbf{A}_k). \tag{18}$$

Here $f_s^{(K)}$ denotes the shared GCN encoder run with K propagation layers; $K = 2$ and $K = 1$ use shared encoder parameters.

The fused structural view is

$$\mathbf{Z}_1 = \mathbf{w} \odot \mathbf{Z}_0 + (\mathbf{1} - \mathbf{w}) \odot \mathbf{Z}_k, \qquad \mathbf{w} = \sigma(\mathbf{Z}_0\mathbf{v}_s + b_s). \tag{19}$$

The base contrastive loss in Eq. (2) is implemented as a symmetrized node-aligned InfoNCE-style objective Oord et al. [2018]. For two view matrices $\mathbf{U}$ and $\mathbf{V}$ with rows $\mathbf{u}_i$ and $\mathbf{v}_i$,

$$\mathcal{C}(\mathbf{U}, \mathbf{V}) = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{\frac{\mathrm{sim}(\mathbf{u}_i, \mathbf{v}_i)}{\tau}}}{\mathcal{Z}_i}, \tag{20}$$

where

$$\begin{aligned}\mathcal{Z}_i = &\sum_{j \neq i} e^{\frac{\mathrm{sim}(\mathbf{u}_i, \mathbf{u}_j)}{\tau}} + \sum_{j=1}^{N} e^{\frac{\mathrm{sim}(\mathbf{u}_i, \mathbf{v}_j)}{\tau}} \\ &+ \sum_{j \neq i} e^{\frac{\mathrm{sim}(\mathbf{v}_i, \mathbf{v}_j)}{\tau}}.\end{aligned} \tag{21}$$

The anchor-wise surrogate in Eq. (4) uses the same positive pairs but keeps only the cross-view in-batch denominator so that each anchor has a clean target-tracking signal.

A.2 Conditional Hypergraph Construction

For each weak anchor $v_i \in \mathcal{V}_{\mathrm{weak}}$, we form a candidate k_h-hop hyperedge

$$e_i^h = \{v_i\} \cup \mathcal{N}_{k_h}(v_i), \qquad \mathcal{N}_{k_h}(v_i) = \{v_j : d(v_i, v_j) \leq k_h\}. \tag{22}$$

We then remove extreme neighbors according to structural-view similarity:

$$\mathcal{N}_{k_h}^{\mathrm{valid}}(v_i) = \{v_j \in \mathcal{N}_{k_h}(v_i) : |s_{ij} - \bar{s}_i| < \beta\varsigma_i\}, \tag{23}$$

where $s_{ij} = \mathrm{sim}(\mathbf{z}_{1,i}, \mathbf{z}_{1,j})$, and $\bar{s}_i$ and ς_i are the mean and standard deviation of candidate-neighbor similarities. The higher-order representation is

$$\mathbf{z}_i^{\mathrm{hyp}} = \sum_{v_j \in \mathcal{N}_{k_h}^{\mathrm{valid}}(v_i)} \gamma_{ij}\mathbf{z}_{1,j}, \qquad \gamma_{ij} = \frac{\exp(s_{ij}/\tau_h)}{\sum_{v_l \in \mathcal{N}_{k_h}^{\mathrm{valid}}(v_i)} \exp(s_{il}/\tau_h)}. \tag{24}$$

We use the same MI-estimation temperature as in Eq. (3), i.e., $\tau_m = \tau$, and set $\tau_h = 2\tau$ so that hypergraph attention is softer than the MI-surrogate estimator Hinton et al. [2015].

Structural controller. The structural supplement is enabled and scaled by a single dataset-level prior scalar. Given the observed non-self edge set E' (or an edge-wise feature/embedding agreement proxy when prior homophily statistics are unavailable), we compute the homophily prior H_y and the controller value α as

$$H_y = \frac{1}{|E'|} \sum_{(i,j)\in E'} \mathbf{1}[y_i = y_j], \qquad \alpha = \mathrm{clip}\left(\frac{H_y - 0.75}{0.05},\, 0,\, 1\right), \tag{25}$$

where the thresholds are fixed across all datasets. The controller value α serves as a dataset-level prior that determines whether, and how strongly, the supplement is used on a given graph. It is computed once before training and is not used by the MI-regulation core (Eqs. (5)– (10)). The supplement is disabled when $\alpha < 0.01$. Otherwise, the effective weak-anchor quantile is $\delta_{\text{eff}} = \alpha\, \delta_{\text{base}}$, where δ_{base} is the nominal quantile; the nominal residual-fusion coefficient and the supplement-loss weight are likewise scaled by α. Weak anchors are selected using r_i in Eq. (11), and the node-wise fusion weight is

$$\bar{s}_i^{\text{val}} = \frac{1}{\left|\mathcal{N}_{k_h}^{\text{valid}}(v_i)\right|} \sum_{v_j \in \mathcal{N}_{k_h}^{\text{valid}}(v_i)} s_{ij}, \tag{26}$$

$$\phi_i = \mathrm{clip}\Big(\alpha\, \phi_{\text{base}}\big[\,0.8 + 0.4\big(1 - \sigma\big(2(\bar{s}_i^{\text{val}} - 0.5)\big)\big)\big],\ 0.05,\ 0.95\Big). \tag{27}$$

Here α controls graph-level activation and scaling, while Eq. (23) filters neighbors by similarity deviation. The bracketed factor in Eq. (27) lies within $[0.8, 1.2]$ and increases as $\bar{s}_i^{\text{val}}$ decreases. It therefore modulates the pre-clipping fusion weight within $[0.8, 1.2]\,\alpha\phi_{\text{base}}$, with the final weight clipped to $[0.05, 0.95]$.

A.3 Training Procedure

Algorithm 1 MIRAGE training procedure

Require: Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, adjacency $\mathbf{A}$, KNN size k, warmup epoch E_w, nominal weak-anchor quantile δ_{base}.

Build the KNN graph $\mathbf{A}_k$ from node attributes.
Precompute m_i^{neigh} from input features and graph neighbors.
Compute α by Appendix A.2.
for $t = 1, \ldots, E$ **do**
  Compute backbone views $(\mathbf{H}_0, \mathbf{H}_1, \mathbf{Z}_0, \mathbf{Z}_1)$ by Eq. (1).
  if $t \geq E_w$ **and** $\alpha \geq 0.01$ **then**
    Compute r_i from $\mathbf{Z}_1$, select $\mathcal{V}_{\text{weak}}$ by Eq. (11), and form $\widetilde{\mathbf{Z}}_1$ by Eq. (12).
    Compute $\mathcal{L}_{\text{sup}}$ by Eq. (13).
  else
    Set $\widetilde{\mathbf{Z}}_1 \leftarrow \mathbf{Z}_1$ and $\mathcal{L}_{\text{sup}} \leftarrow 0$.
  end if
  Compute bounded setpoints $\mu_i^{(b)}$ from $p_i^{(0,b)}$; the setpoint head and encoder are jointly trainable.
  Compute $\mathcal{L}_{\text{base}}$ and anchor-wise surrogates $\widehat{I}_i^{(b)}$ using $\widetilde{\mathbf{Z}}_1$.
  Compute $\mathcal{L}_{\text{alloc}}$, $\mathcal{L}_{\text{ideal}}$, $\mathcal{L}_{\text{view}}$, and $\mathcal{L}_{\text{reg}}$ using $\widetilde{\mathbf{Z}}_1$, as specified in Eqs. (6)– (10).
  Minimize $\mathcal{L} = \mathcal{L}_{\text{base}} + \mathcal{L}_{\text{reg}} + \mathbb{I}[t \geq E_w]\, \mathcal{L}_{\text{sup}}$.
end for
return $\mathbf{E} = \mathbf{H}_0 + \mathbf{H}_1 + \widetilde{\mathbf{Z}}_1$.

A.4 Complexity of the Regulation Operator

For a sampled node set $\mathcal{B}$, computing the anchor-wise surrogate in Eq. (4) costs $O(|\mathcal{B}|^2 d)$ due to the in-batch similarity matrix. Full-graph training sets $|\mathcal{B}| = N$, while sampled training reduces

this term by using a negative pool. The structural supplement is applied only to $|\mathcal{V}_{\text{weak}}|$ anchors and costs $O(|\mathcal{V}_{\text{weak}}|\bar{k}_h d)$ after k_h-hop candidate neighbors have been collected, where $\bar{k}_h$ is the average number of valid neighbors after filtering. The KNN graph is constructed once before training as part of the backbone view instantiation.

A.5 Practical Tuning

We tune MIRAGE as three grouped blocks rather than as several unrelated objectives: (i) λ_{MI} is the main knob; (ii) λ_{dev}, λ_{var}, λ_{hyper}, and η are kept small and swept as a grouped auxiliary scale; (iii) the warmup epoch E_w delays the structural supplement until the base contrastive views have stabilized. The core-only model is obtained by disabling the structural supplement, including both residual fusion and its auxiliary loss.

B Conceptual Comparison with Closely Related Architectures

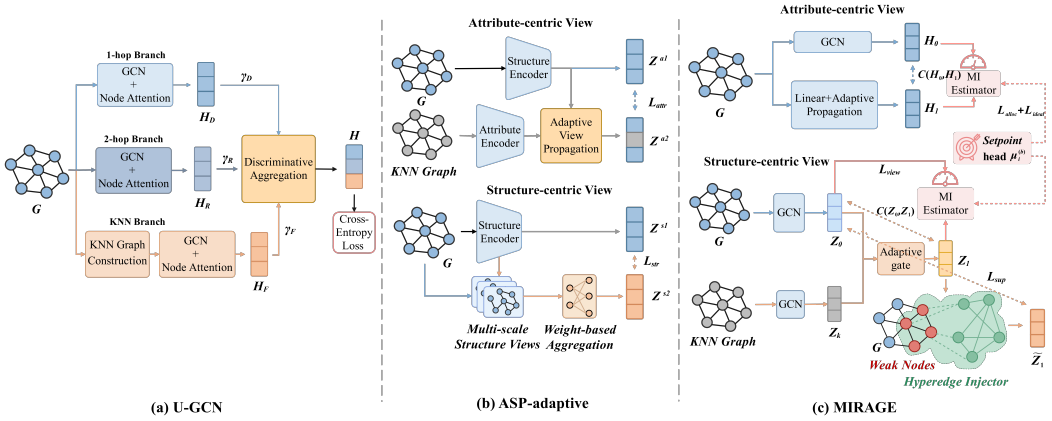


Figure 5: Comparison of graph representation learning architectures. (a) U-GCN aggregates 1-hop, 2-hop, and kNN branches for discriminative structural learning Jin et al. [2021]. (b) ASP-adaptive dynamically integrates node attributes with multi-scale structural views Chen et al. [2025]. (c) MIRAGE uses a target-tracked MI-surrogate regulation operator: anchor-wise setpoints, branch-level stabilization, and a conditional structural supplement.

C Hyperparameters and Notation

To facilitate a clear conceptual understanding of the MIRAGE framework, Table 4 summarizes the key mathematical notations and parameter symbols used throughout our methodology. This glossary serves as a quick reference to outline the essential variables and coefficients that define our formulation. Table 5 further reports the corresponding hyperparameter settings and validation ranges used in our experiments.

Table 4: Glossary of key mathematical notations.

Symbol	Meaning	Symbol	Meaning
τ	Contrastive temperature	δ	Weak-anchor quantile threshold
τ_m	MI-surrogate temperature, set to τ	τ_h	Hypergraph attention temperature
$\mu_{\min}, \mu_{\max}$	Bounds for anchor-wise setpoints	$\mu_i^{(b)}$	Anchor-wise setpoint for branch b
$\mathbf{v}_\mu, b_\mu$	Trainable setpoint-head parameters	λ_{ideal}	Penalty weight for deviation from the MI target.
λ_1	Attribute-branch loss weight	λ_{MI}	Allocation-loss weight
λ_{dev}	Branch-level deviation penalty	λ_{var}	Across-anchor dispersion penalty
λ_{hyper}	Supplement stability weight	$\mathcal{B}$	Batch or negative-pool node set
ϕ_{base}	Nominal residual-fusion coefficient	ϵ	Supplement target offset
β	Neighbor-similarity outlier filter	w_i	Degree-based loss weight

Table 5: Hyperparameter settings and validation ranges.

Group	Hyperparameter	Setting / validation range
MI setpoints	$[\mu_{\min}, \mu_{\max}]$	default $(0.6, 0.8)$; sensitivity: $(0.3, 0.5)$, $(0.6, 0.8)$, $(0.9, 1.1)$
Setpoint head	$\mathbf{v}_\mu, b_\mu$	trainable
MI loss	λ_{MI}	$\{0.1, 0.3, 0.5\}$, selected by validation
Stability	$\lambda_{\mathrm{dev}}, \lambda_{\mathrm{var}}$	defaults 0.009 and 0.001; grouped scale $\{0.5, 1.0, 2.0\}$
Supplement	$\lambda_{\mathrm{hyper}}, \eta$	grouped with auxiliary scale
Supplement	ϕ_{base}	$\{0, 0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 1.0\}$
Supplement	β	1.5 for neighbor-deviation filtering
Temperatures	τ_h	$2\tau_m$ with $\tau_m = \tau$ in this implementation
Schedule	E_w	selected by validation; supplement disabled before warmup
Evaluation	repeats	10 random seeds; mean $\pm$ standard deviation reported
Reference term	λ_{ideal}	0.5; 0 in fixed-target runs

D Datasets and Implementation Details

We consider seven graph datasets and characterize their structural properties to inform conditional structural supplementation in MIRAGE.

- **Cora and Citeseer.** Sen et al. [2008] Cora and Citeseer are widely used citation network benchmarks for collective classification. We follow the standard Planetoid splits. The Cora dataset consists of 2,708 machine learning papers, while Citeseer contains 3,327 scientific publications, with citation links defining the graph structure.
- **Wiki-CS.** Mernyei and Cangea [2020] Wiki-CS is a node classification benchmark derived from the computer science subset of Wikipedia. Articles are assigned to 10 high-level research areas (e.g., computational linguistics, operating systems, and computer security) after cleaning and consolidating the original category hierarchy. To mitigate the sensitivity of graph neural networks to data splits, Wiki-CS provides 20 predefined random train/validation/test splits, which are commonly used to assess robustness in semi-supervised learning settings.
- **Amazon-Photo.** McAuley et al. [2015] Amazon-Photo is a widely adopted node classification benchmark constructed from the Amazon co-purchase network. Nodes correspond to photography-related products, and edges indicate that two products are frequently purchased together. Node labels are determined by product categories, and node features are represented using bag-of-words encodings of product reviews.
- **ACM.** Wang et al. [2019], Yang et al. [2024a] ACM is a heterogeneous information network benchmark comprising three types of nodes: papers, authors, and subjects. The task is to classify paper nodes according to their research areas based on publication venues. Node features are bag-of-words representations of paper keywords. Structural relationships are defined using PAP (paper–author–paper) and PSP (paper–subject–paper) meta-paths.
- **Cora-ML.** Bojchevski and Günnemann [2017] Cora-ML is a classic citation network benchmark and a curated subset of the larger Cora dataset, originally introduced in Deep Gaussian Embedding of Graphs: Unsupervised Inductive Learning via Ranking. It is constructed as a directed attributed graph, where nodes represent papers and features are bag-of-words encodings of abstracts. This dataset is commonly used to evaluate models on unsupervised node classification under directed graph settings. For Amazon-Photo, ACM, and Cora-ML, we adopt random splits with 10% training, 10% validation, and 80% testing nodes Yang et al. [2025].
- **Minesweeper.** Platonov et al. [2023] Minesweeper is a synthetic benchmark consisting of 10,000 nodes arranged in a regular 100×100 grid, inspired by the classic Minesweeper game. The task is a binary classification problem to identify randomly placed "mine" nodes. Node features are one-hot encodings of the number of neighboring mines. For Minesweeper, we follow the 10 official predefined splits provided by prior work Platonov et al. [2023].

Table 6: Statistics of the six primary benchmarks and the additional Minesweeper diagnostic dataset.

Dataset	# Nodes	# Edges	# Labels	# Features	Homophily
Cora	2,708	5,278	7	1,433	0.8100
Citeseer	3,327	4,552	6	3,703	0.7355
WikiCS	11,701	215,603	10	300	0.6543
Amazon-Photo	7,650	119,081	8	745	0.8272
ACM	3,025	13,128	3	1,870	0.8208
Cora-ML	2,995	8,158	7	2,879	0.7886
Minesweeper	10,000	39,402	2	7	0.6828

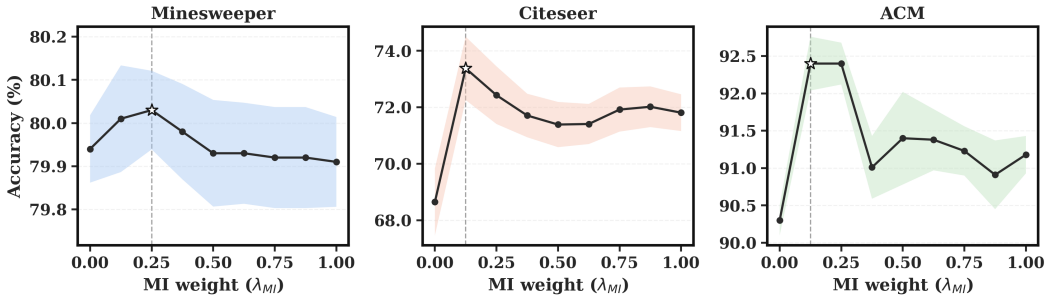


Figure 6: Parameter sensitivity of the MI-regulation weight λ_{MI} on three datasets.

Implementation and evaluation protocol. We implement MIRAGE using PyTorch Geometric Fey and Lenssen [2019]. Experiments are conducted on an NVIDIA A40 GPU. We follow a linear evaluation protocol: after representation training, node embeddings are L_2-normalized and evaluated with logistic regression. The regularization parameter is chosen by grid search with 5-fold cross-validation. Each experiment is repeated over 10 random seeds; we report mean accuracy and standard deviation.

E Additional Experimental Results

E.1 Sensitivity to the MI-Regulation Weight

Fig. 6 shows the effect of the MI-regulation weight λ_{MI} on Minesweeper, Citeseer, and ACM. The trend supports the grouped-sensitivity result in the main paper: moderate MI-surrogate regulation is beneficial, while overly weak or overly strong regularization can reduce accuracy.

E.2 Sensitivity to the Hypergraph Residual Weight

Table 7 reports the effect of the maximum residual fusion weight ϕ_{base}. The best values on ACM and Cora-ML are near 0.25, and larger values gradually hurt performance. This supports the interpretation that the structural supplement should be residual and gated, not an always-dominant branch.

E.3 Coarse Sweep of MI Interval and Hypergraph Warmup

Fig. 7 presents a coarse sweep over wider MI intervals and warmup epochs (evaluated with the hypergraph path active across all datasets to inspect injection timing). It is useful as an exploratory diagnostic: moderate intervals tend to perform best, while overly tight or loose intervals can either over-constrain informative anchors or admit too many noisy anchors. The result also shows that delayed supplement activation is often preferable to immediate injection.

Table 7: Impact of the hypergraph residual weight ϕ_{base} on node classification accuracy (%).

ϕ_{base}	ACM	Cora-ML
0.000	92.56 ± 0.41	85.88 ± 0.42
0.125	92.57 ± 0.42	85.90 ± 0.43
0.250	$\mathbf{92.63 \pm 0.28}$	$\mathbf{85.99 \pm 0.42}$
0.375	92.43 ± 0.42	85.98 ± 0.42
0.500	92.58 ± 0.32	85.96 ± 0.52
0.625	92.40 ± 0.16	85.97 ± 0.44
0.750	92.54 ± 0.23	85.88 ± 0.35
0.875	92.30 ± 0.34	85.80 ± 0.28
1.000	92.27 ± 0.35	85.79 ± 0.28

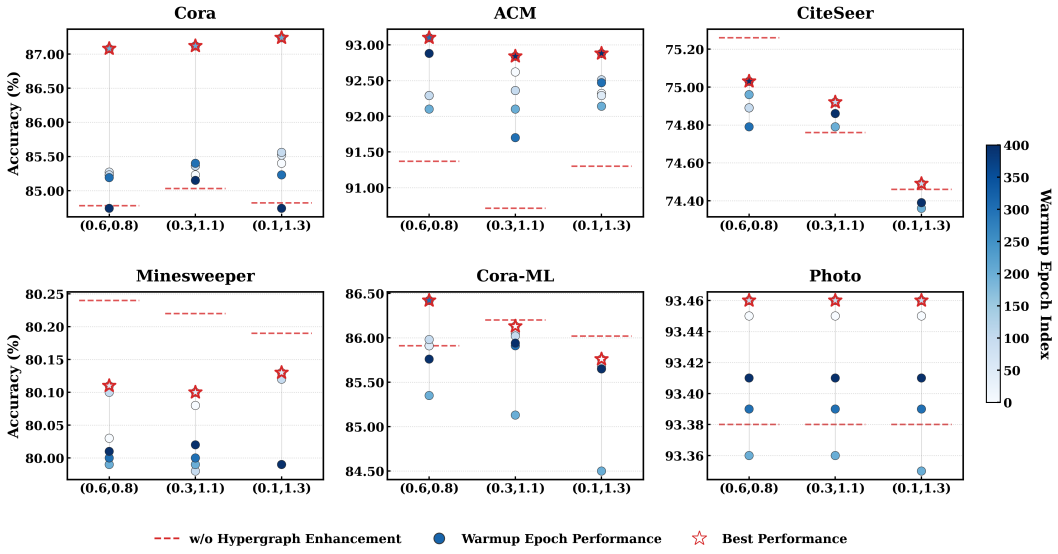


Figure 7: Coarse sweep of adaptive MI intervals and hypergraph warmup. Each point corresponds to a warmup epoch; red dashed lines denote the no-supplement baseline and red stars mark the best configuration.

E.4 Focused Sweep of MI Interval and Hypergraph Warmup

Fig. 8 reports a focused sweep of MI intervals and hypergraph warmup settings. Across the shown benchmarks, the default interval $(0.6, 0.8)$ typically balances selective structural supplementation with noise control. On Citeseer and Minesweeper, the no-supplement baseline can be as good as or better than enhanced variants, reinforcing the claim that the structural path is conditional rather than universally beneficial.

E.5 Perturbation Stability of View-Level Regulation

To examine view-level stabilization under structural perturbations, we compare MIRAGE variants with and without this component on Cora (Table 8). Stabilization substantially reduces variance without noise and yields lower standard deviations at four of five nonzero noise levels, complementing the MI-tracking evidence in Fig. 3.

E.6 Hypergraph Gate Diagnostics

Table 9 compares three variants: MIRAGE-core disables the conditional structural path; MIRAGE-full (always-on) enables it without the reliability gate; MIRAGE-gate uses the structure-reliability gate. On Cora and ACM, the gated variant is better than always-on aggregation, indicating that

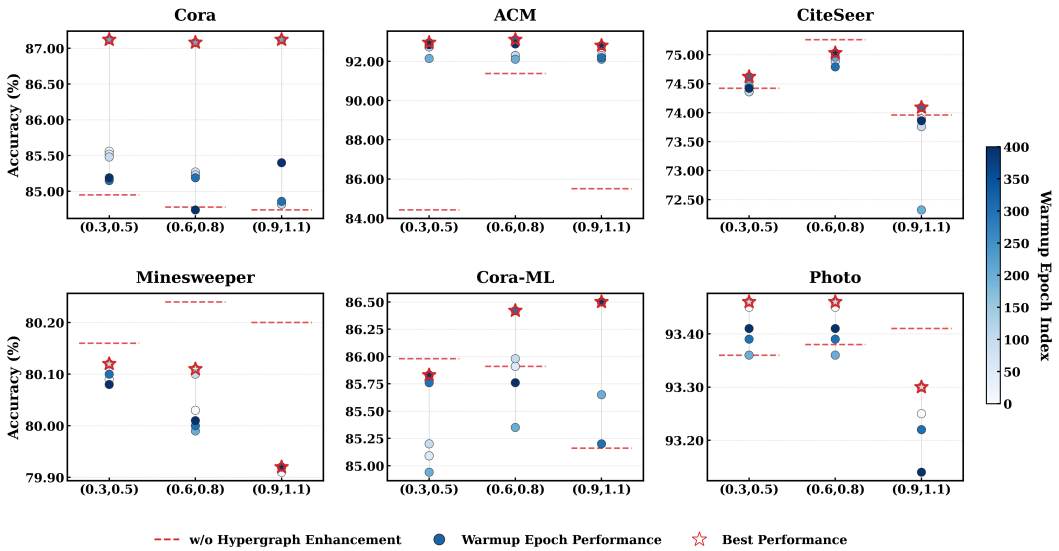


Figure 8: Focused sweep of adaptive MI target interval and hypergraph warmup strategy. Each point corresponds to a warmup epoch setting; the red dashed line is the model without hypergraph enhancement and the star marks the best-performing configuration identified by grid search.

Table 8: Effect of view-level MI-surrogate stabilization under stochastic structural perturbations on Cora.

Noise	w/ View-level Stabilization	w/o View-level Stabilization
0	85.21 ± 0.24	84.84 ± 0.89
0.02	85.02 ± 0.83	84.91 ± 0.80
0.04	84.96 ± 0.77	85.00 ± 0.82
0.06	85.10 ± 0.64	85.00 ± 0.82
0.08	85.11 ± 0.64	84.98 ± 0.82
0.10	85.11 ± 0.65	85.03 ± 0.83

indiscriminate higher-order aggregation injects noise even when the graph is largely homophilous. On Citeseer, the gate matches core-only and improves over always-on, showing that it behaves conservatively when the supplement would be unhelpful.

Table 9: Effect of hypergraph gating strategy on node classification accuracy (%).

Dataset	MIRAGE-core	MIRAGE-full (always-on)	MIRAGE-gate
Cora	84.82 ± 0.10	84.63 ± 0.60	$\mathbf{85.02 \pm 0.13}$
Citeseer	$\mathbf{73.95 \pm 0.84}$	73.68 ± 0.69	$\mathbf{73.95 \pm 0.84}$
ACM	91.32 ± 0.09	90.67 ± 0.44	$\mathbf{92.26 \pm 0.21}$

E.7 Progressive MI-Regulation Core Attribution

Table 10 decomposes the MI-regulation core. The node-level allocation loss and the view-level stabilization loss are individually dataset-dependent, but their combination gives a coherent core model.

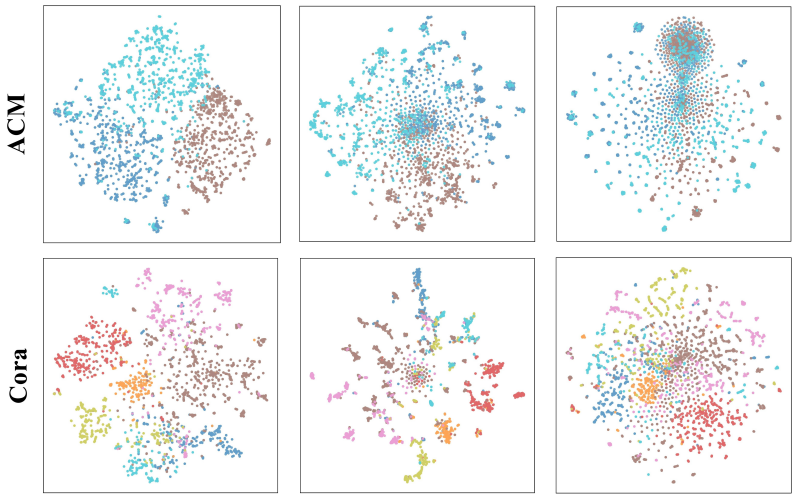
E.8 Qualitative Visualization

Fig. 9 visualizes representations on ACM and Cora. This figure is included only as qualitative intuition; the main claims are supported by the quantitative and mechanism-level analyses above.

Table 10: Progressive ablation of MI-regulation components (%).

Variant	Citeseer	ACM
V1: Base ($\mathcal{L}_{\text{base}}$ only)	73.11 ± 0.52	91.75 ± 0.49
V2: $+\mathcal{L}_{\text{alloc}} + \mathcal{L}_{\text{ideal}}$	73.88 ± 0.86	91.53 ± 0.08
V3: $+\mathcal{L}_{\text{view}}$	73.06 ± 0.47	91.99 ± 0.55
V4: Core ($\mathcal{L}_{\text{alloc}} + \mathcal{L}_{\text{ideal}} + \mathcal{L}_{\text{view}}$)	73.95 ± 0.84	91.32 ± 0.09
Full MIRAGE	$\mathbf{73.95 \pm 0.84}$	$\mathbf{92.26 \pm 0.21}$

MIRAGE produces compact clusters relative to ML^2-GCL and iGCL, which is consistent with the target-allocation interpretation but should not be treated as standalone evidence.



（a）MIRAGE （b）ML²-GCL （c）iGCL

Figure 9: t-SNE visualization of learned representations on ACM and Cora.

F Extended Limitations

Our evaluation focuses on linear probing for node classification. Link prediction, graph classification, few-shot evaluation, and large-scale graph pretraining remain future work. Full-graph MI-surrogate computation uses pairwise similarities and is therefore quadratic in the number of nodes unless negative sampling or approximation is used. The current datasets are mostly moderate-to-high homophily; Minesweeper is a stress case but not a sufficient evaluation of strong heterophily. The conditional structural supplement assumes that higher-order neighborhoods contain useful semantic information, which can fail when neighborhoods are unreliable. Finally, MIRAGE is more complex than single-loss GCL baselines. Our grouped sensitivity analysis and practical tuning recipe reduce the tuning burden, but they do not remove it.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims in the abstract and introduction are consistent with the experimental results and theoretical analysis, accurately describing the scope and limitations of the paper's contributions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper explicitly discusses the assumptions, scope, and potential shortcomings of the method in a dedicated Limitations section.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theorems are fully proven in the appendix, with intuitive sketches provided in the main text.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes model architecture, hyperparameter settings, and data processing pipelines in detail in the main text and appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: The implementation code is not publicly released at this stage, as it is being retained for ongoing research and further methodological development. To support reproducibility, the paper and appendix provide detailed descriptions of the proposed method and experimental settings.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: Experimental settings including data splits, learning rate, and optimizer are detailed in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: All main results report mean and standard deviation across multiple random seed runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The appendix includes the details on the computer resources.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have reviewed the NeurIPS Code of Ethics and the research complies in all respects.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This work can positively impact downstream applications such as knowledge graph construction and community detection on citation and social networks.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The models/datasets released in this work do not pose high-risk misuse scenarios.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators and original owners of all datasets used in this work are properly cited in the references.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This work does not publicly release new datasets, model checkpoints, or implementation code. The proposed method and its experimental evaluation are documented in the paper and appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This research does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This research does not involve human subjects and therefore does not require IRB review.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method of MIRAGE consists of anchor-wise MI setpoints and dual-view MI stabilization, with no LLMs involved as a methodological component.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.