International Journal of Approximate Reasoning 193 (2026) 109657

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

journal homepage: www.elsevier.com/locate/ijar

HALO: Hardness-aware bilevel-inspired contrastive graph clustering

Yuchen Zhu [a,b], Kuang Zhou [a,c,*], Haishan Ye [b,d,*], Guang Dai [b], Ivor W. Tsang [e,f]

[a] School of Mathematics and Statistics, Northwestern Polytechnical University, 710129, Xi'an, PR China
[b] SGIT AI Lab, State Grid Corporation of China, PR China
[c] MOE Key Laboratory for Complexity Science in Aerospace, Northwestern Polytechnical University, 710129, Xi'an, PR China
[d] Center for Intelligent Decision-Making and Machine Learning, School of Management, Xi'an Jiaotong University, 710049, Xi'an, PR China
[e] College of Computing and Data Science, Nanyang Technological University, Singapore
[f] Agency for Science, Technology and Research (A*STAR), Singapore

ARTICLE INFO

Keywords:
Uncertainty
Uncertain clustering
Deep clustering
Hard samples

ABSTRACT

Attributed graph clustering is a pivotal task in network analysis, yet it is often hindered by inherent uncertainty arising from structural noise and attribute ambiguity. To address this, contrastive graph clustering has emerged as a powerful paradigm, primarily owing to its effectiveness in refining decision boundaries through hard sample mining. However, current methods define hardness primarily via instance-level similarity, a metric that overlooks the cluster-level uncertainty of structurally ambiguous nodes. For samples with conflicting neighborhoods or attributes, this uncertainty renders their representations unreliable. Despite their critical role in achieving robust clustering, these high-uncertainty samples are consistently overlooked in existing frameworks. To overcome this limitation, we propose HALO, a Hardness-Aware biLevels-inspired cOntrastive clustering framework. Specifically, HALO extracts two cluster-level structural signals as two views and couples them with instance-level similarity to form a bilevel hardness measure. This measure identifies hard samples whose embedding behaviors are inconsistent across levels–an indicator of clustering ambiguity–and adaptively amplifies their contribution during learning. Theoretically, we show that HALO's gradient dynamics naturally magnify uncertain regions in the embedding space, while an optimal-transport-based alignment ensures consistent clustering distributions across views and prevents degenerated solutions. Extensive experiments on various real datasets demonstrate that HALO discovers substantially more informative hard pairs, presents the generalization ability, and achieves state-of-the-art clustering performance.

1. Introduction

Graph clustering has become a fundamental task in network analysis, aiming to partition nodes into distinct groups based on both structures and attributes [1]. Despite its progress, graph clustering is frequently plagued by inherent uncertainty arising from noisy structures and ambiguous attributes, which obscures the boundaries between clusters. For example, epistemic uncertainty arises

* Corresponding authors.
E-mail addresses: h1nkik@mail.nwpu.edu.cn (Y. Zhu), kzhoumath@nwpu.edu.cn (K. Zhou), yehaishan@xjtu.edu.cn (H. Ye), guang.gdai@gmail.com (G. Dai), ivor_tsang@cfar.a-star.edu.sg (I.W. Tsang).

https://doi.org/10.1016/j.ijar.2026.109657
Received 28 November 2025; Received in revised form 31 January 2026; Accepted 26 February 2026
Available online 2 March 2026

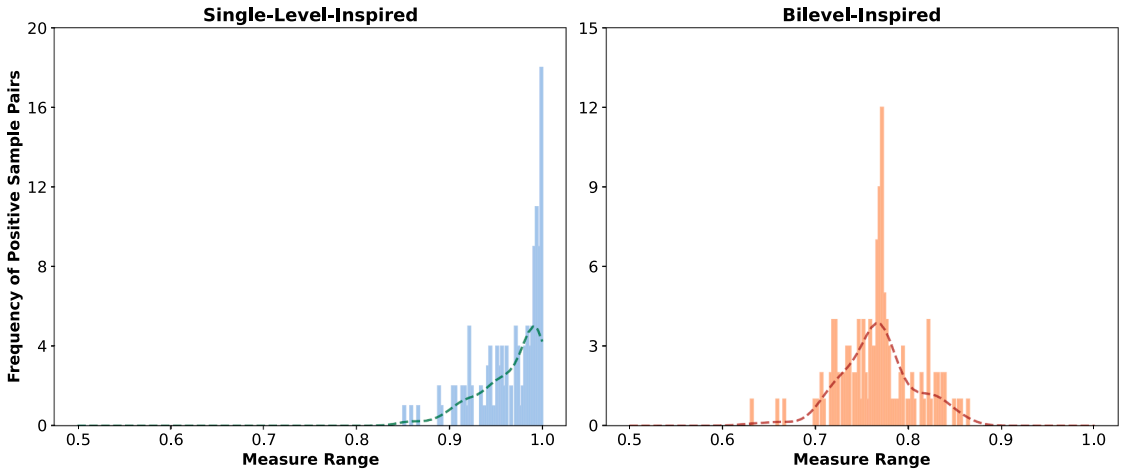


Fig. 1. Two histograms of positives from the Brazil Air-Traffic data via different measures. More examples can be found in Section 4.7.

when structurally proximal nodes–such as those sharing common neighbors–belong to distinct clusters, whereas aleatoric uncertainty is typically induced by inherent attribute noise or feature overlap.

To address these issues, advanced frameworks have been developed. For instance, CDC [2] adaptively learns high-quality anchors to mitigate the uncertainty caused by random selection, while AORLNet [3] introduces interpretable attribute weighting to reduce feature ambiguity. Parallel to these advancements, contrastive learning has emerged as a potential powerful approach to tackle these challenges because of its simplicity and ease of understanding [4,5]. By specifically targeting and learning from these hard instances, including hard positives (*i.e.*, semantically similar but embedding-wise distant) and hard negatives (*i.e.*, semantically dissimilar but embedding-wise close), this technique effectively refines decision boundaries and mitigates the adverse effects of uncertainty [6]. In the context of contrastive clustering, sample hardness serves as an observable manifestation of representation uncertainty, and is explicitly quantified via sample pair similarity: samples that exhibit high similarity despite being negative pairs (or low similarity despite being positive) are identified as hard instances.

However, most existing contrastive graph clustering methods employ an instance-level contrastive loss (such as NT-Xent [7]) that measures similarity via cosine distance [8–10]. These single-level metrics tend to equate geometric proximity in the embedding space with sample hardness, overlooking the semantic structures that emerge at the cluster level. As a result, they cannot faithfully capture the representational uncertainty of nodes with ambiguous cluster memberships, making it difficult to identify truly informative hard pairs. We take the Brazil Air-Traffic (BAT) dataset as an illustrative example (Fig. 1). Under a single-level metric (left), most positive pairs cluster in the high-similarity region, yielding a long-tailed distribution. These seemingly "hard" samples are actually trivial to distinguish and thus contribute little to optimization. In contrast, our bilevel-inspired metric (right), which integrates cluster-level semantics from the soft-assignment head, reshapes the distribution into a mixture-like form with heavier mass in moderate-similarity regions. This indicates that the model now captures more uncertain yet informative positive pairs–those with ambiguous cluster memberships that were previously overlooked. By explicitly modeling such uncertainty, the learned representation becomes more discriminative and robust for downstream clustering.

To mitigate this, several studies introduce external guidance–*e.g.,* selection modules or mixture models–to refine density estimation and mine hard samples [11,12]. While partially effective, such approaches suffer from two limitations: (i) generalization is tightly coupled to auxiliary components, so end performance inherits their biases and failure modes; and (ii) these modules are not tailored for clustering objectives, causing a misalignment between learned representations and downstream partitioning [13].

Motivated by these challenges, we ask a fundamental question: *can internal knowledge alone provide a more principled alternative to single-level metrics–one that also reveals truly informative hard samples through uncertainty?* To this end, we propose HALO, a uncertainty-guided, end-to-end contrastive graph clustering method. HALO derives cluster-aware signals from the shared encoder and constructs a bilevel-inspired hardness that bridges instance-level similarities with cluster-level semantics, thereby turning uncertainty into a supervisory signal for both hard positives and hard negatives.

In addition, most contrastive clustering methods [14–16] adopt a Siamese pipeline [7] that generates two augmented views and aligns transposed assignment matrices. To enhance interpretability, we provide an equivalent formulation that performs the alignment via optimal transport with a symmetrized loss, and we theoretically justify its validity. An equipartition constraint is further incorporated to prevent degeneracy during alignment.

In sum, the main contributions of this work are as follows:

- We present a simple yet effective contrastive graph clustering framework whose representations are guided by uncertainty-aware, cluster-level signals derived internally from the shared encoder, ensuring clustering-friendly embeddings in an end-to-end manner.
- From a gradient perspective, we show that instance-only NT-Xent loss underutilizes uncertainty-rich regions, whereas HALO's bilevel hardness reallocates gradient mass toward informative hard positives and negatives, enabling unified and effective hard-sample mining.

- We provide an equivalent view-alignment scheme via optimal transport and a symmetrized objective, theoretically justifying consistency across augmentations; an equipartition constraint prevents degenerate solutions.
- Extensive experiments on public benchmarks demonstrate generalization ability and consistent improvements over related baselines, with ablations and analyses offering insights into HALO's uncertainty-aware behavior.

The remainder of this paper is organized as follows. In Section 2, we introduce some related works. Section 3 provides a detailed understanding of the proposed method. In Section 4, performance analysis through extensive experiments is reported. In Section 5, we conclude the paper.

2. Related works

In this section, we review the methodologies of the uncertain clustering and progress of hard samples mining (HSM).

2.1. Uncertain clustering

Uncertainty-aware clustering can be broadly categorized into decision-uncertainty and representation-uncertainty paradigms, depending on where the ambiguity is modeled during the clustering process. In detail, the decision-uncertainty refers to the uncertain assignment of samples according to the raw data features, while the representation-uncertainty characterizes the unreliability of the learned representations themselves, by estimating uncertainty over latent embeddings or predictive evidence. In terms of decisions, classical fuzzy clustering [17,18] assigns each sample to a set of continuous memberships, reflecting its partial affiliation to multiple clusters. Evidential clustering [19,20] further generalizes this idea by introducing belief mass functions and credal partitions to explicitly represent committed, partial, and vacuous uncertainty. Such approaches are effective in identifying ambiguous or boundary samples, *e.g.*, GFDC [21] detects outliers across clusters of heterogeneous densities, and MvWECM [22] aggregates multi-view evidence into consistent cluster beliefs. In addition, three-way clustering [23], inspired by three-way decision theory, incorporates an explicit indeterminate region between acceptance and rejection, allowing samples in boundary or overlapping areas to be placed into a "deferment" state rather than forced into a single cluster. However, this decision-uncertainty paradigm may require a huge computing capacity with increasing clusters and stronger evidence. In contrast, representation-uncertainty methods model uncertainty in the latent space. The notable framework is evidential deep learning (EDL) [24], which offers an alternative way to model representation-uncertainty by treating deep network outputs as Dirichlet evidence [25]. Building on this idea, several recent deep clustering methods combine EDL-style uncertainty with contrastive learning [26,27], typically by assigning Dirichlet distributions to pairwise similarities or pseudo-label predictions. However, with respect to these approaches aimed for images data, uncertainty is still defined at the instance level, without explicitly incorporating cluster-level structural information (*e.g.*, soft cluster assignments or higher-order relations among clusters); as a result, they struggle to capture structural ambiguities such as boundary-crossing samples or inconsistent assignments across clusters. Moreover, because the evidential regularization is imposed on a single embedding space without an explicit mechanism to control the interaction between clustering structure and uncertainty, these methods are prone to trivial solutions that all samples look like equally uncertain or equally certain. By contrast, our method treats uncertainty as a cluster-aware hardness signal: we derive a bilevel hardness measure from cluster-level information. On top of that, we add an equal-partition constraint on assignments that inherently mitigates degenerated solutions, preventing all samples to be assigned to one certain cluster.

2.2. Hard sample mining

Positive and negative samples measure plays a pivotal role in the performance of contrastive learning methods. For clustering tasks, Zhao *et al.* [28] addressed the bias in only negative samples through a special sampling strategy. More specific, two-component beta mixture model in ProGCL [12] is fit on the feature similarity to recognize true hard negative samples, but it degrades when meeting the case that data is concentrated at the boundaries as Fig. 1 (left) shows. Zhu *et al.* [29] recognized hard boundary samples with assistance of pseudo labels generated by K-means. Nevertheless, most of these approaches depend on external priors – such as pre-trained embeddings or clustering labels – to estimate sample hardness. Such dependence limits their generalization ability, as the downstream clustering performance directly hinges on the reliability of these prior models. For example, K-means is unsuitable for non-Euclidean data, and the resulting pseudo labels often lack consistency for guiding deep representation learning. In this paper, we take a different perspective: leveraging internal cluster knowledge to infer hardness signals from model gradients, yielding a second-order measure that captures the representation-uncertainty.

3. Methodology

In this section, we provide a detailed introduction of our clustering method, whose overall architecture is illustrated in Fig. 2. We first familiarize readers with notations that will be used throughout the paper in Section 3.1. After revisiting the prior art in Section 3.2, we propose a novel hard samples mining technique inspired by cluster- and instance-level metrics in Section 3.3. The clustering alignment and algorithm summary are presented in Sections 3.4 and 3.5, respectively.

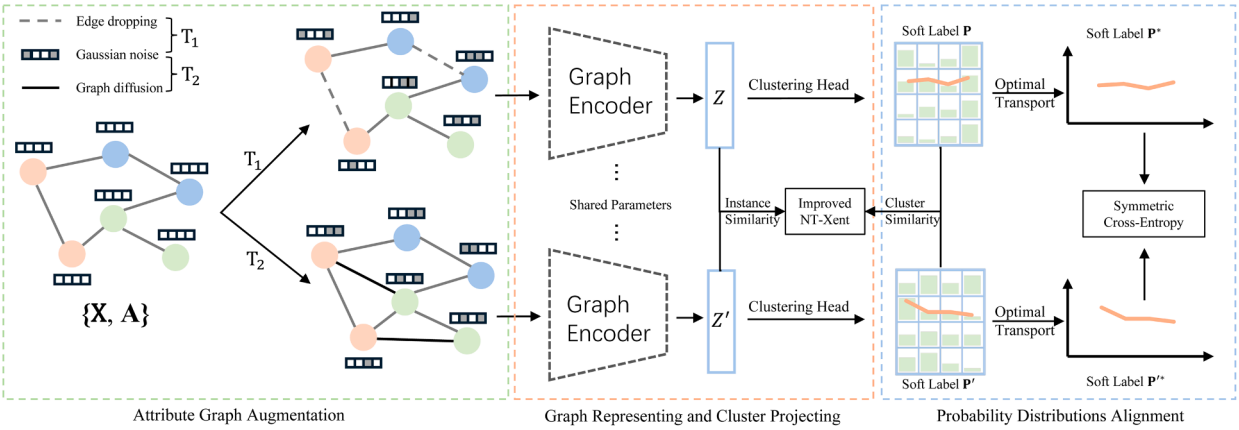


Fig. 2. Overview of our proposed HALO. Two augmented views of the graph $\mathcal{G} = \{\mathbf{X}, \mathbf{A}\}$ are processed by different operators T_1 and T_2, which are encoded in a Siamese network including GCNs and a clustering head. The soft assignments of the two views, $\mathbf{P}$ and $\mathbf{P}'$, are obtained from the embeddings $\mathbf{Z}$ and $\mathbf{Z}'$, respectively. By aligning two views of soft labels $\mathbf{P}^*$ and $\mathbf{P}'^*$ after the optimal transport, HALO produces consistent clustering results.

3.1. Preliminaries and notations

Given an undirected graph $\mathcal{G} = \{\mathcal{E}, \mathcal{V}\}$, where $\mathcal{E}$ and $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ are the edge set and node set, respectively. We denote the number of nodes as $N = |\mathcal{V}|$. $\mathbf{X} \in \mathbb{R}^{N \times D}$ is the node attribute matrix and D is the raw attribute dimension. $\mathbf{A} \in \mathbb{R}^{N \times N}$ is the adjacent matrix of $\mathcal{G}$, where $\mathbf{A}_{ij} = 1$ denotes that there exists a connection between node v_i and node v_j; otherwise, $\mathbf{A}_{ij} = 0$.

Data Augmentation Graph data augmentation is crucial for contrastive learning representation. Two common techniques [30,31], attribute corruption and edge perturbation, are used in our method. In the attribute level, we input a random noise matrix $\mathbf{N} \in \mathbb{R}^{N \times D}$ from a Gaussian distribution $\mathcal{N}(1, 0.1)$. The resulting corrupted attribute matrix $\widetilde{\mathbf{X}} \in \mathbb{R}^{N \times D}$ can be formulated as

$$\widetilde{\mathbf{X}} = \mathbf{X} \odot \mathbf{N}, \tag{1}$$

where $\odot$ denotes the Hadamard product. In the structure level, we consider to employ the edge dropping on one view, and graph diffusion on another view. For the edge dropping, we have

$$\mathbf{A}^m = \widetilde{\mathbf{D}}^{-\frac{1}{2}}((\mathbf{A} \odot \mathbf{M}) + \mathbf{I})\widetilde{\mathbf{D}}^{-\frac{1}{2}}, \tag{2}$$

where $\mathbf{I} \in \mathbb{R}^{N \times N}$ and $\widetilde{\mathbf{D}}$ are an identity matrix and the degree matrix of $\mathbf{A} + \mathbf{I}$, respectively. And $\mathbf{M} \in \mathbb{R}^{N \times N}$ is a masked matrix, where some first-order linkage relations are removed randomly. For the graph diffusion, the normalized adjacency matrix is transformed to a graph diffusion matrix by Personalized PageRank [32]:

$$\mathbf{A}^n = \alpha\left(\mathbf{I} - (1-\alpha)\left(\widetilde{\mathbf{D}}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\widetilde{\mathbf{D}}^{-\frac{1}{2}}\right)\right)^{-1}, \tag{3}$$

where α is the teleport probability. Finally, we have two augmented views of the raw graph $\mathcal{G}$: $(\widetilde{\mathbf{X}}, \mathbf{A}^m)$ and $(\widetilde{\mathbf{X}}, \mathbf{A}^n)$.

Graph Encoder Graph convolutional network (GCN) is powerful in aggregating the first-order neighbor information for representing embeddings. The aggregation function is formulated as follows:

$$\mathbf{Z}^{(l)} = \sigma\left(\widetilde{\mathbf{D}}^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I})\widetilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{Z}^{(l-1)}\boldsymbol{W}^{(l)}\right), \tag{4}$$

where $\mathbf{Z}^{(l)} \in \mathbb{R}^{N \times d^{(l)}}$ and $\boldsymbol{W}^{(l)} \in \mathbb{R}^{d^{(l-1)} \times d^{(l)}}$ denote the latent representations and network parameters of the l_{th} layer, respectively. σ denotes the Tanh activation function. In this work, we apply a three-layer ($l = 2$) GCN to construct the encoder f, where $\mathbf{Z}^{(0)} = \mathbf{X}$. The Siamese network (parameters-sharing) is utilized since it is a natural tool for comparing entities [33]. Hence, denote $\mathbf{Z} = f\left(\widetilde{\mathbf{X}}, \mathbf{A}^m; \boldsymbol{W}\right)$ and $\mathbf{Z}' = f\left(\widetilde{\mathbf{X}}, \mathbf{A}^n; \boldsymbol{W}\right)$ be two embeddings learned from two augmented graphs.

3.2. The prior art of contrastive loss

Recently, contrastive methods have become the main stream in the domains of graph clustering. The critical idea is to improve the discriminativeness of features by pulling together the positive samples while push away the negative ones. Benefiting from normalized embeddings and an extra proper temperature parameter, the normalized temperature-scaled cross-entropy [7] (NT-Xent) becomes prevalent in graph clustering tasks [14,34] due to its simplicity and efficiency. Let $\boldsymbol{z}_i$ and $\boldsymbol{z}'_j$ denote i_{th} sample of $\mathbf{Z}$ and j_{th} sample of $\mathbf{Z}'$, respectively. Then, the similarity of $\boldsymbol{z}_i$ and $\boldsymbol{z}'_j$ can be defined as [35]

$$S'_{ij} \triangleq \boldsymbol{z}_i^\top \boldsymbol{z}'_j / \|\boldsymbol{z}_i\| \|\boldsymbol{z}'_j\|. \tag{5}$$

Without loss of generality, given the similarity matrix $\boldsymbol{S} = (S'_{ij})_{N\times N}$, the NT-Xent style in clustering is formulated as

$$\mathcal{L}(\boldsymbol{S}) = -\frac{1}{N}\sum_{i=1}^{N}\log\frac{e^{S'_{ii}/\tau}}{\sum_{j=1,j\neq i}^{N} e^{S'_{ij}/\tau}}, \tag{6}$$

where $\tau > 0$ is the temperature parameter. Eq. (6) indicates that two views of embeddings from the same node are considered as the positive samples pair. The rest of $(N-1)$ pairs of different nodes are treated as negative ones.

Before discussing about the drawback of $\mathcal{L}$, we first dive into a useful gradient framework of similarity metric learning. In brief, the General Pair Weighting (GPW) [36] framework provides a unified perspective and a powerful tool for understanding sample pairs-based loss functions through gradient analysis, where gradient dynamics reflect the optimization process of representations, offering a complementary perspective to the objective value [3,37,38]. GPW allows us to interpret how the loss function implicitly weights different sample pairs via gradients.

For the general loss $\mathcal{L}$ with the parameters $\boldsymbol{W}$ of deep neural network, the derivative at a certain iteration is calculated as

$$\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial\boldsymbol{W}} = \frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial\boldsymbol{S}}\frac{\partial\boldsymbol{S}}{\partial\boldsymbol{W}} = \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}}\frac{\partial S'_{ij}}{\partial\boldsymbol{W}}. \tag{7}$$

In fact, we can use a new function $\mathcal{F}$ whose gradient w.r.t $\boldsymbol{W}$ is exactly the same as Eq. (7) to transform into a new equation. $\mathcal{F}$ is formulated as

$$\mathcal{F}(\boldsymbol{S}) = \sum_{i=1}^{N}\sum_{j=1}^{N}\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}}S'_{ij}. \tag{8}$$

For the two augmented views without labels, we rewrite Eq. (8) as below:

$$\begin{aligned}\mathcal{F}(\boldsymbol{S}) &= \sum_{i=1}^{N}\left(\sum_{j=1,j\neq i}^{N}\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}}S'_{ij} + \frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ii}}S'_{ii}\right)\\ &= \sum_{i=1}^{N}\left(\sum_{j=1,j\neq i}^{N} w_{ij}S'_{ij} - w_{ii}S'_{ii}\right),\end{aligned} \tag{9}$$

where $w_{ij} = \left|\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}}\right|$. Therefore, learning with $\mathcal{L}$ is transformed from Eq. (7) into learning weights of Eq. (9). It is crucial that the assumptions $\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}} \geq 0$ and $\frac{\partial\mathcal{L}(\boldsymbol{S})}{\partial S'_{ii}} \leq 0$ should be satisfied within the GPW framework. Within this framework, we find that the NT-Xent style loss treats the hard positive sample pairs equally to the easy positive ones, hindering learning the high-quality discriminative space. The formal statement is as follows. Its proof can be found in the Appendix A.

Proposition 1. *The loss function $\mathcal{L}$ can mine hard negatives, but it treats the hard positives equally to the easy ones. That is, their weights are the same constant.*

The gradient serves as the medium for seeking optimal solutions. If the loss function only includes a first-order metric (*e.g.*, cosine similarity), the information from this metric vanishes during the gradient calculation in the parameter update process. As a result, all sample pairs have an identical impact on the gradient, making it impossible to distinguish their contributions effectively.

3.3. Cluster- and instance-level metrics-inspired HSM

To capture uncertainty and sample hardness in unsupervised representation learning, we introduce a hardness-aware mining mechanism inspired by both cluster-level and instance-level metrics. Assume that there are C clusters. Denote the soft labels as $p(c_k|\boldsymbol{z}_i) = p_{ik}$ and $p(c_k|\boldsymbol{z}'_j) = p_{j'k}$, indicating that the probability of the i_{th} and j_{th} instance from the two augmented views being assigned to the k_{th} cluster, respectively. These probabilistic assignments serve as learnable uncertainty-aware prototypes, generated by a two-layer projection head $f_C : \mathbb{R}^{N\times d} \to \mathbb{R}^{N\times C}$ followed by a Softmax activation:

$$\mathbf{P} = (p_{ik})_{N\times C} = \text{Softmax}(f_C(\mathbf{Z})). \tag{10}$$

Cluster-level Uncertainty The cluster-level discrepancy between two views can be quantified as $Q'_{ijk} = \left|p_{ik} - p_{j'k}\right|$. It follows the law of attraction and repulsion of clustering: If two embeddings belong to the same k_{th} cluster, then $\left|p_{ik} - p_{j'k}\right|$ is close to 0; If two embeddings belong to different cluster, then $\left|p_{ik} - p_{j'k}\right|$ is close to 1. Summing over all clusters, $\sum_{k=1}^{C} Q'_{ijk}$ represents a global cluster-level dissimilarity supported by all prototypes. When this sum approaches C, it implies that the sample pair exhibits strong disagreement across clusters–an indicator of epistemic uncertainty and potential hardness.

Intuitively, this cluster-level inconsistency serves as a proxy for uncertainty regarding the decision boundary. Consider a sample located deep within a cluster manifold (an “easy" sample); its structural and attribute patterns are robust, leading to consistent cluster assignments across different views despite perturbations. Conversely, a sample located at the decision boundary or in an overlapping region (a “hard" sample) is highly sensitive to perturbations. Different augmentations may push it toward different cluster centroids,

resulting in a divergence between p_{ik} and $p_{j'k}$. Therefore, a high value of $\sum_{k=1}^{C} Q'_{ijk}$ explicitly captures this boundary ambiguity, identifying samples that are difficult for the model to categorize confidently.

Unified Hardness Measurement To combine the global cluster-level cue with the local instance-level similarity $S'_{ij} = \langle z_i, z'_j \rangle$, we define a hardness factor that balances the attraction-repulsion dynamics between instances:

$$\begin{aligned} \alpha'_{ij} &= \sum_{k=1}^{C} (Q'_{ijk} + \frac{\xi}{C}) - \chi_{ij} S'_{ij} \\ &= \sum_{k=1}^{C} \left| p_{ik} - p_{j'k} \right| - \chi_{ij} S'_{ij} + 2, \end{aligned} \tag{11}$$

where $\chi_{ij} = 1$ iif $i = j$ and $\chi_{ij} = -1$ otherwise, and we set $\xi = 2$. This formulation offers two appealing properties: (1) it jointly encodes cluster-level uncertainty ($\sum_k Q'_{ijk}$) and instance-level similarity (S'_{ij}); (2) it is differentiable w.r.t S'_{ij} and nonnegative–facilitating smooth hardness reweighting during learning. larger values of α'_{ij} indicate more uncertain or conflicting samples, which deserve stronger emphasis in training.

Incorporating α'_{ij} into contrastive learning yields a second-order hardness-weighted loss:

$$\mathcal{L}_{\text{int}}(\boldsymbol{S}) = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\tilde{\alpha}'_{ii} S'_{ii}/\tau}}{\sum_{j=1, j\neq i}^{N} e^{\tilde{\alpha}'_{ij} S'_{ij}/\tau}}, \tag{12}$$

where $\tilde{\alpha}'_{ij} = \frac{\alpha'_{ij}}{\|\alpha'_{ij}\|}$. The temperature parameter τ is fixed to 0.1 throughout this paper. For convenience, we term it INT-Xent (improved NT-Xent). Comparing with the conventional loss, INT-Xent is able to recognize hard sample pairs. Formally, a proposition is presented below and its proof is attached in the Appendix B.

Proposition 2. *The INT-Xent can mine the hard sample pairs. Concretely, it up-weights more dissimilar pairs of positive samples and more similar pairs of negative ones.*

This property is particularly valuable in the absence of ground truth, where no ground truth exists to directly label samples as "hard" or "easy." Following [6], hardness in clustering can only be inferred at the pair level, not for individual instances. Therefore, our metric pair (S'_{ij}, Q'_{ijk}) provides a principled means to approximate pairwise hardness: S'_{ij} captures local similarity, while Q'_{ijk} introduces a global uncertainty prior. By coupling them, INT-Xent prevents the model from overfitting to "easy" positives and mitigates false-negative bias, thereby yielding a more uncertainty-robust representation space.

3.4. Clustering distribution alignment

After obtaining the soft labels from the two augmented views, we now perform the final step of clustering – aligning the predicted distributions. Each soft label vector $\mathbf{P}$ or $\mathbf{P}'$ can be seen as an uncertain belief about how likely each instance belongs to every cluster. Because these two beliefs come from different perturbations of the same graph, they may disagree, especially for ambiguous samples. Therefore, it is reasonable to make one view learn from the other through a cross-entropy loss, which encourages the second view to be calibrated toward the first one. The cross-entropy loss is formulated as

$$\ell(p_{ik}, p_{i'k}) = -\sum_{k=1}^{C} p_{ik} \log p_{i'k}. \tag{13}$$

To avoid a trivial solution occurring in optimization that all of instances are allocated to a single cluster, the equipartition constraint which means to assign samples into clusters uniformly is added in the cross-entropy. We attempt to transport the current distributions $\mathbf{P}$ and $\mathbf{P}'$ to more uniform distributions $\mathbf{P}^*$ and $\mathbf{P}'^*$ with less information cost. Considering the following constrained optimization problem:

$$\begin{aligned} &\min_{p,p'} -\sum_{i=1}^{N} \sum_{k=1}^{C} p_{ik} \log p_{i'k} \\ &\text{s.t.} \sum_{i=1}^{N} p_{ik} = \frac{N}{C}. \end{aligned} \tag{14}$$

Notice that there is an implicit condition $\sum_{k=1}^{C} p_{ik} = 1$ in Eq. (14) because of the Softmax. All the ways of transporting are formally denoted as

$$\mathcal{P} := \{\mathbf{P} \in \mathcal{R}_{+}^{N\times C} \mid \mathbf{P}\mathbf{1}_C = \mathbf{1}_N, \mathbf{P}^{\top}\mathbf{1}_N = \frac{N}{C}\mathbf{1}_C\}, \tag{15}$$

where $\mathbf{1}_N$ and $\mathbf{1}_C$ are the row vector of all ones in dimension of N and C, respectively. Furthermore, Eq. (14) can be rewritten as a following optimal transport problem:

$$\min_{\mathbf{P}\in\mathcal{P}} \langle \mathbf{P}, -\log \mathbf{P}' \rangle, \tag{16}$$

where $\langle \cdot, \cdot \rangle$ is the Frobenius inner product. The fast version of Sinkhorn-Knopp algorithm [39] is used to solve the transport problem (16) effectively with a $1/\eta$ regularization, which is a simple iterative method aiming to the transportation polytope (15). In the end, a more uniform distribution $\mathbf{P}^*$ is given by

$$\mathbf{P}^* = \mathrm{diag}(a)(\mathbf{P}')^{\eta}\mathrm{diag}(b), \tag{17}$$

where a and b are two renormalization vectors and the exponentiation is element-wise. The algorithm process is summarized in the Appendix C. Since there is no ground truth to tell which view is more accurate, we finally adopt a dual alignment strategy:

$$\mathcal{L}_{\mathrm{clus}} = \sum_{i=1}^{N} \left[\ell(p^*_{ik}, p^*_{i'k}) + \ell(p^*_{i'k}, p^*_{ik}) \right]. \tag{18}$$

The resulting symmetrical cross-entropy thus reflects a form of mutual trust between the two uncertain views, helping the model become more stable and less sensitive to noise. In addition, the alignment consistency of probability distributions between two views can be guaranteed by the following theorem.

Theorem 1. *Minimizing the clustering loss (18) makes the distributions $\mathbf{P}^*$ and $\mathbf{P}'^*$ of two views tend to be consistent.*

The proof can be found in Appendix D. Since the soft label distributions of the two views tend to be the same, we can take the average as a final assignment, which also avoids a few error bias. Therefore, the final clustering prediction is calculated by

$$\hat{y}_i = \arg\max_k \frac{1}{2}(p_{ik} + p_{i'k}). \tag{19}$$

Finally, we arrive at the overall objective function of HALO, which lies in the simple form of

$$\mathcal{L}_{\mathrm{overall}} = \mathcal{L}_{\mathrm{int}} + \lambda \mathcal{L}_{\mathrm{clus}}, \tag{20}$$

where λ is a trade-off hyperparameter that can be tuned. In practice, it can be set to 0.1 by default.

3.5. Algorithm

Our method is summarized in Algorithm 1. Recall the previous mathematical notations. N, C and d are the number of nodes, clusters and dimensions, respectively. The computing complexity (per epoch) of the two-levels measure is $\mathcal{O}(C^2N + N^2d)$, and the computing complexity of Sinkhorn is $\mathcal{O}(NCt)$, where t is the iteration of Sinkhorn. However, it could be ignored due to $Ct < Nd$. Therefore, the total time complexity of the overall loss is $\mathcal{O}(C^2N + N^2d)$.

Algorithm 1 HALO.

Input: The graph $\mathcal{G} = \{\mathbf{X}, \mathbf{A}\}$, the number of clusters C, the maximum number of epoch I.
Initialization: iteration $i = 0$.
Output: The final clustering assignment.

repeat
  Calculate the corrupted attribute matrix $\widetilde{\mathbf{X}}$ via Eq. (1).
  Calculate the masked adjacent matrix $\mathbf{A}^m$ via Eq. (2).
  Calculate the graph diffusion matrix $\mathbf{A}^n$ via Eq. (3).
  Calculate graph representations of $\{\widetilde{\mathbf{X}}, \mathbf{A}^m\}$ and $\{\widetilde{\mathbf{X}}, \mathbf{A}^n\}$ via Eq.(4).
  Calculate probability assignments $\mathbf{P}, \mathbf{P}'$ via Eq. (10).
  Calculate constrained assignments $\mathbf{P}^*, \mathbf{P}'^*$ via the fast sinkhorn algorithm.
  Calculate the overall loss $\mathcal{L}_{\mathrm{overall}}$ via Eq. (20).
  Update parameters by the Adam and back-propagation algorithm.
  $i \leftarrow i + 1$.
until $i \geq I$.
Calculate predicted cluster labels via Eq. (19).

4. Experiments

In this section, we evaluate the proposed HALO on six real graph clustering benchmarks compared with several related methods. A series of comparisons, ablation studies, and so on are carried out to investigate properties of the method.

4.1. Experimental setup

We implement HALO on the server equipped with the NVIDIA A100 (80GB) and an Intel(R) Xeon(R) Platinum 8358 Processor based on the PyTorch platform. All results are obtained under five runs with different random seeds. In terms of configurations, the probability of removing edges is fixed to 0.1 following DCRN [30]. For the Siamese network, the graph encoder is initialized by the xavier initialization [40]. The temperature of INT-Xent is $\tau = 0.1$. Code is available at: https://github.com/H1nkik/HALO.

Table 1
The detailed statistics of the datasets. The symbol $\bar{d}$ denotes the average node degree, calculated as $2|\mathcal{E}|/N$, and h represents the homophily ratio, defined as the fraction of edges connecting nodes with the same label.

Dataset	Nodes	Edges	Feature Dims	Cluster Numbers	Average Degree ($\bar{d}$)	Homophily Ratio (h)
ACM	3,025	13,128	1,870	3	8.68	0.82
EAT	399	5,994	203	4	30.04	0.40
UAT	1,190	13,599	239	4	22.86	0.70
Cornell	183	149	1,703	5	1.89	0.29
Amazon	24,492	93,050	300	5	7.60	0.38
Squirrel	5,201	217,073	2,089	5	76.27	0.22

4.2. Datasets

To evaluate the performance of HALO, we apply it to six widely-used graph clustering datasets, including EAT, UAT, Cornell, ACM, Squirrel and Amazon [41–43].

- EAT (Traffic): Data collected from the Statistical Office of the European Union (Eurostat) from January to November 2016. Airport activity is measured by the total number of landings plus takeoffs in the corresponding period.
- UAT (Traffic): Data collected from the Bureau of Transportation Statistics from January to October, 2016. Airport activity is measured by the total number of people that passed (arrived plus departed) the airport in the corresponding period.
- Cornell (Webpage): It is a subset of WebKB that is a webpage dataset collected from computer science departments of various universities by Carnegie Mellon University. Nodes represent web pages, and edges are hyperlinks between them. Node features are the bag-of-words representation of web pages. The web pages are manually classified into the five categories, student, project, course, staff, and faculty.
- ACM (Citation): This is a paper network from the ACM dataset. There is an edge between two papers if they are written by same author. Paper features are the bag-of-words of the keywords. Selected papers are divide into three classes (database, wireless communication, data mining) by their research area.
- Amazon (Item): This dataset is based on the Amazon product co-purchasing network metadata dataset from SNAP Datasets. Nodes are products (books, music CDs, DVDs, VHS video tapes), and edges connect products that are frequently bought together.
- Squirrel (Webpage): It is a page-page network on specific topics in Wikipedia. Nodes represent web pages and edges are mutual links between pages. Node features correspond to several informative nouns in the Wikipedia pages.

The detailed information of these datasets is shown in Table 1. The average degree $\bar{d}$ is given by

$$\bar{d} = \frac{1}{|\mathcal{V}|}\sum_{i=1}^{|\mathcal{V}|} d_i = \frac{2|\mathcal{E}|}{N}, \tag{21}$$

where d_i is the degree of the i_{th} node. A higher value implies denser interactions between nodes, providing stronger topological support for representation learning. In contrast, graphs with low average degree are structurally sparse, causing neighborhood information to become unreliable and increasing uncertainty in the learned embeddings. The homophily ratio h measures the proportion of edges linking nodes of the same class. It is defined as

$$h = \frac{\left|\left\{(v_i, v_j) \in \mathcal{E} \mid y_i = y_j\right\}\right|}{|\mathcal{E}|}, \tag{22}$$

where $Y = \left\{y_i, 1 \le i \le N\right\}$ is the true label. High-homophily graphs exhibit clear community structures, where structural cues directly correlate with cluster assignments. Conversely, low-homophily graphs possess predominantly cross-class edges, making structural signals misleading and substantially increasing the difficulty of cluster separation. Such datasets stress-test the robustness and generalization capability of deep clustering models.

4.3. Evaluation

To evaluate the clustering methods, four common metrics are applied [19,22]: accuracy (ACC), normalized mutual information (NMI), adjusted rand index (ARI), and F1-score (F1). These metrics reflect the degree of consistency, similarity and alignment between ground-truths and predicted labels. Generally, higher values of these metrics indicate better clustering performance.

Formally, let $\hat{Y} = \left\{\hat{y}_i, 1 \le i \le n\right\}$ denote the predicted labels. ACC is defined as

$$\text{ACC} = \frac{1}{N}\sum_{i=1}^{N} \delta\big(\text{map}\left(\hat{y}_i\right), y_i\big), \tag{23}$$

where we utilize the Kuhn-Munkres algorithm [44] to determine the optimal mapping $\text{map}(\cdot)$ between cluster assignments and ground truths. NMI can be computed as follows:

$$\text{NMI}(\hat{Y}, Y) = \frac{2MI(\hat{Y}, Y)}{H(\hat{Y}) + H(Y)}, \tag{24}$$

where $H(\cdot)$ and $MI(\cdot)$ are the entropy and mutual information, respectively. ARI is formulated as

$$\mathrm{ARI} = \frac{\mathrm{RI} - \mathbb{E}[\mathrm{RI}]}{\max(\mathrm{RI}) - \mathbb{E}[\mathrm{RI}]}, \tag{25}$$

where RI indicates the Rand Index. F1 is the harmonic mean of Precision and Recall.

4.4. The compared state-of-the-arts

To validate the effectiveness of our method, we compare with 10 state-of-the-art deep graph clustering methods:

- GK is a combination of graph encoder and K-means. We first use a graph encoder which is the same as HALO to produce a embedding space. Then we perform K-means on it.
- MVGRL [45] consists of two dedicated GNNs and graph pooling layers (Siamese architecture). It maximizes the mutual information between node representations and graph representations between views.
- AGCN [46] designs a heterogeneity-wise fusion module to dynamically fuse the node attribute and the topological structure feature, and it includes a scale-wise fusion module to adaptively aggregate the multi-scale features embedded at different layers.
- AGC-DRR [15] makes contrast based on the Siamese network with an adversarial learning mechanism. And it applies a dual redundancy reduction strategy that decreases the information redundancy.
- ProGCL [12] utilizes the Beta mixture model to estimate the uncertainty of a negative sample, and performs K-means on the learned embeddings.
- Dink-Net [47] scales to large datasets effectively using mini-batch technique and running a discriminative pre-text task. In an adversarial manner, it minimizes its proposed cluster dilation loss and cluster shrink loss.
- FDAGC [48] incorporates with the concept of fuzzy clustering in an end-to-end manner without additionally conventional clustering techniques.
- HCHSM [11] leverages a hierarchically contrastive scheme within Siamese architecture that can collect multilevel graph information for hard sample comparison, which can select better hard samples for representation learning from the mutual information estimation perspective.
- MAGI [49] proposes a community-aware graph contrastive learning framework that uses modularity maximization in mini-batch form as its pretext task and avoids semantic drift.
- HCAGC [50] explores and distinguishes the pseudo homophily within Siamese architecture, and it well-designs a triplet self-supervised clustering objective for refined representations of hard samples.

For fairness, all compared methods are implemented using the same settings as described in the original papers or codes. We also conduct a search for the optimal configurations following their guidelines. For example, we search the parameters lambda3 and lambda4 of FDAGC ranging from $[10^1, 10^2, 10^3, 10^4, 10^5]$ and $[0.0005, 0.005, 0.05, 0.5, 5]$, respectively. In addition to the parameters of models, we also try different training parameters such as learning rate in $[0.0001, 0.0005, 0.01]$ of MAGI.

4.5. Performance results

As can be observed in Table 2, several key observations can be made as follows: Firstly, the generalization ability of existing methods is limited. For example, AGC-DRR performs well on the citation network (ACM) but degrades significantly on traffic graphs (EAT and UAT). Similarly, Dink-Net presents a good result on the item network (Amazon) rather than the citation network (ACM). Compared to them, although HALO merely exhibit marginal improvements on every dataset, it consistently ranks first or second across various graph types, showing an impressive degree of generalization. Secondly, we observe that methods with explicit HSM strategies, such as HCHSM and HCAGC, often achieve stronger performance compared to those methods without considering the HSM. However, these outstanding approaches heavily depend on manually designed hardness estimators or sampling heuristics, which may overfit to domain-specific structural patterns. In contrast, HALO maintains competitive or even superior results without relying on external mining modules, implying that our hardness-aware objective inherently captures informative and difficult samples in a more adaptive manner. This observation further highlights the importance of handling hard samples in a principled and uncertainty-aware way for generalizable graph clustering. Additionally, we also analyze the empirical runtime and memory of baselines in Appendix E.

4.6. Ablation study

In this subsection, we conduct ablation studies to empirically validate the effectiveness of each component in our model, including INT-Xent, alignment and the backbone setting. The overall loss Eq. (20) consists of two parts, the INT-Xent marked as “H" and clustering alignment marked as “C". Concretely, “w/o H" indicates the INT-Xent is replaced by the traditional NT-Xent. “w/o C" means that we only apply one best view to output cluster labels instead of view fusion. From the ablation results in Fig. 3, both “H" and “C" are essential, each contributing significantly to performance improvement. For instance, there are around 50% decrease of F1 “w/o H" in ACM and around 15% decrease of ACC “w/o C" in EAT. Therefore, "H" contributes to discriminative representation learning, while "C" ensures reliable clustering consistency, and together they form the foundation of HALO’s generalization ability. The backbone experiments are shown in Appendix F.

Table 2
The average clustering performance and standard deviation of five runs on real datasets are reported. The best results are highlighted in **bold**, and the runner up underlined. "-" and "OOM" indicate failure to produce results and raise the out-of-memory error, respectively.

Dataset	Metric	GK	AGCN	MVGRL	AGC-DRR	Dink-Net	MAGI	HCAGC	ProGCL	FDAGC	HCHSM	HALO
ACM	ACC	56.16 ± 1.55	87.49 ± 0.22	40.02 ± 0.87	87.55 ± 0.39	44.91 ± 5.68	82.26 ± 1.91	84.27 ± 1.29	55.26 ± 2.90	82.53 ± 4.33	58.73 ± 5.76	**87.65 ± 0.81**
	NMI	29.98 ± 1.31	60.04 ± 0.42	1.44 ± 0.24	60.23 ± 1.19	7.94 ± 6.58	55.09 ± 1.80	52.99 ± 1.31	27.89 ± 4.57	53.12 ± 3.23	30.50 ± 13.60	**60.45 ± 1.46**
	ARI	36.43 ± 1.22	66.12 ± 0.63	1.51 ± 0.16	65.74 ± 1.29	7.78 ± 6.42	60.92 ± 1.54	56.56 ± 1.35	20.78 ± 5.83	55.98 ± 3.15	29.05 ± 11.49	**66.21 ± 1.65**
	F1	57.87 ± 1.81	87.08 ± 0.27	32.81 ± 3.65	86.86 ± 0.50	34.98 ± 5.29	85.23 ± 1.72	83.40 ± 1.54	45.07 ± 2.73	82.66 ± 3.49	47.42 ± 4.96	**87.23 ± 0.81**
EAT	ACC	21.21 ± 1.96	36.09 ± 0.95	33.03 ± 0.64	35.39 ± 1.21	26.07 ± 0.00	39.04 ± 2.19	49.49 ± 1.70	39.98 ± 10.80	45.29 ± 1.20	32.98 ± 0.12	**50.07 ± 1.32**
	NMI	2.82 ± 2.02	8.50 ± 1.18	11.21 ± 0.48	8.50 ± 1.03	0.00 ± 0.00	8.80 ± 2.22	**32.41 ± 2.07**	15.88 ± 10.64	19.34 ± 0.62	10.80 ± 0.14	19.93 ± 1.80
	ARI	1.55 ± 1.10	4.14 ± 0.56	4.46 ± 0.53	4.53 ± 0.56	0.00 ± 0.00	3.80 ± 1.37	18.06 ± 1.91	10.73 ± 8.71	16.97 ± 1.36	3.95 ± 0.11	**18.06 ± 1.58**
	F1	13.63 ± 1.70	29.63 ± 1.07	25.77 ± 0.53	29.28 ± 1.35	11.24 ± 0.00	29.42 ± 1.07	48.98 ± 2.41	30.72 ± 15.49	46.46 ± 2.32	25.64 ± 0.18	**49.10 ± 1.91**
UAT	ACC	30.58 ± 4.29	37.90 ± 0.00	45.49 ± 1.41	45.16 ± 2.31	35.80 ± 0.00	45.66 ± 2.50	**51.60 ± 1.01**	40.40 ± 4.41	47.49 ± 1.24	48.43 ± 1.31	50.74 ± 0.52
	NMI	10.77 ± 3.79	13.36 ± 0.02	22.90 ± 0.80	15.65 ± 2.42	8.81 ± 0.00	19.29 ± 1.65	23.07 ± 1.46	22.24 ± 1.28	18.89 ± 2.00	7.07 ± 7.75	**23.42 ± 0.96**
	ARI	5.50 ± 3.77	6.50 ± 0.00	16.79 ± 1.39	11.06 ± 1.38	5.14 ± 0.00	12.27 ± 2.20	18.91 ± 1.24	13.61 ± 1.82	16.73 ± 1.33	3.54 ± 8.07	**20.35 ± 0.35**
	F1	25.60 ± 4.42	26.74 ± 0.00	41.41 ± 2.19	39.58 ± 3.42	23.98 ± 0.00	45.51 ± 2.07	47.13 ± 0.98	33.53 ± 4.96	44.01 ± 1.54	25.58 ± 11.06	**47.32 ± 0.64**
Cornell	ACC	31.59 ± 2.00	–	40.98 ± 3.61	32.35 ± 1.36	40.84 ± 0.91	41.39 ± 1.11	35.52 ± 1.58	42.90 ± 0.82	45.88 ± 1.02	**47.84 ± 0.44**	43.50 ± 1.75
	NMI	3.12 ± 0.79	–	3.24 ± 0.39	4.28 ± 0.82	2.26 ± 0.11	2.80 ± 0.41	4.14 ± 1.10	4.39 ± 0.94	1.06 ± 0.73	3.18 ± 0.13	**4.47 ± 1.08**
	ARI	0.88 ± 0.54	–	0.02 ± 1.69	−0.19 ± 1.18	−1.55 ± 0.40	0.69 ± 0.89	0.21 ± 1.21	3.50 ± 3.70	−0.02 ± 0.59	−0.42 ± 0.16	**5.39 ± 1.36**
	F1	15.27 ± 1.34	–	20.97 ± 0.67	22.97 ± 1.58	20.99 ± 0.31	18.47 ± 1.74	24.36 ± 2.00	21.61 ± 1.55	17.59 ± 1.73	20.19 ± 0.41	**25.08 ± 2.12**
Amazon	ACC	28.17 ± 1.02	32.58 ± 0.97	OOM	26.27 ± 0.00	30.87 ± 0.02	26.42 ± 0.35	26.11 ± 0.88	–	OOM	27.65 ± 0.82	**35.17 ± 0.40**
	NMI	0.38 ± 0.15	0.25 ± 0.02		1.13 ± 0.00	1.21 ± 0.00	0.19 ± 0.10	0.99 ± 0.28	–		**1.60 ± 0.08**	1.25 ± 0.18
	ARI	0.45 ± 0.07	0.42 ± 0.04		1.10 ± 0.00	**1.63 ± 0.01**	0.42 ± 0.09	0.21 ± 0.30	–		1.61 ± 0.13	0.00 ± 0.22
	F1	15.64 ± 0.91	19.26 ± 0.46		22.11 ± 0.00	22.47 ± 0.01	19.88 ± 0.71	20.18 ± 0.66	–		22.40 ± 0.33	**23.51 ± 0.59**
Squirrel	ACC	21.24 ± 0.89	23.45 ± 0.97	25.04 ± 0.15	25.22 ± 0.54	21.47 ± 1.16	23.92 ± 0.56	25.02 ± 0.91	21.43 ± 0.36	23.54 ± 0.25	24.60 ± 0.18	**25.87 ± 0.45**
	NMI	1.17 ± 0.70	0.82 ± 0.40	1.89 ± 0.15	1.59 ± 0.03	0.46 ± 0.12	0.76 ± 0.10	1.80 ± 0.33	0.74 ± 0.39	0.99 ± 0.14	1.67 ± 0.00	**1.91 ± 0.19**
	ARI	0.81 ± 0.38	0.65 ± 0.36	0.77 ± 0.03	0.99 ± 0.05	1.09 ± 0.20	0.42 ± 0.15	1.02 ± 0.20	0.07 ± 0.00	0.47 ± 0.16	0.65 ± 0.00	**1.41 ± 0.18**
	F1	15.29 ± 1.04	13.46 ± 0.56	17.84 ± 0.99	21.36 ± 0.93	10.75 ± 2.42	23.75 ± 0.42	22.23 ± 0.50	10.71 ± 0.85	18.81 ± 0.42	17.07 ± 0.00	**25.64 ± 0.51**

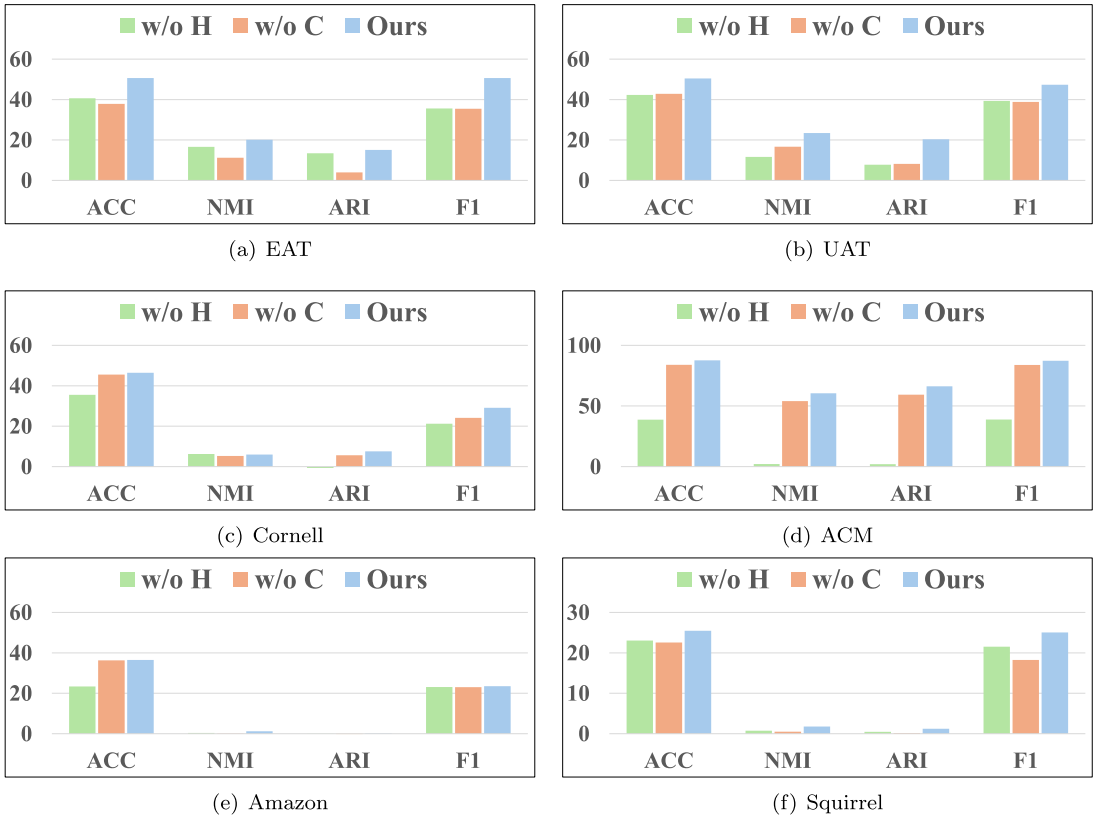
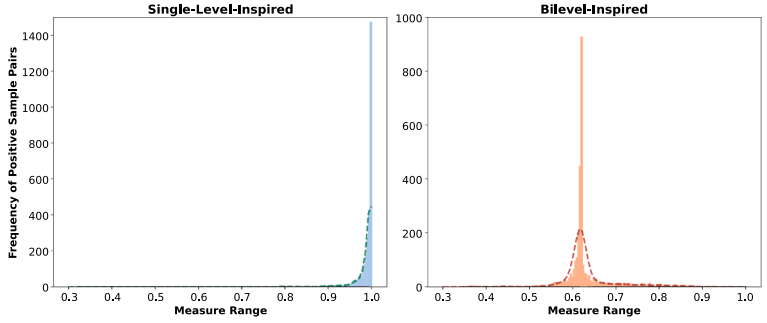
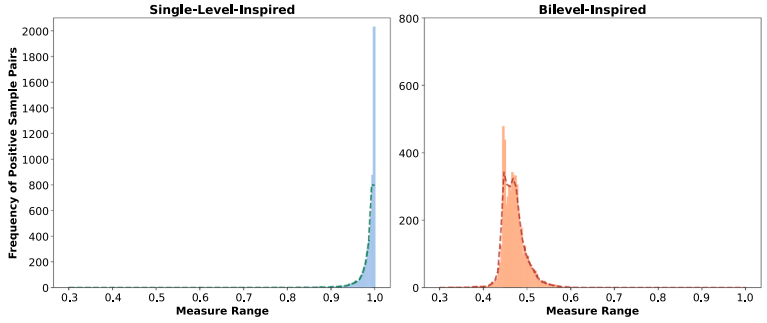


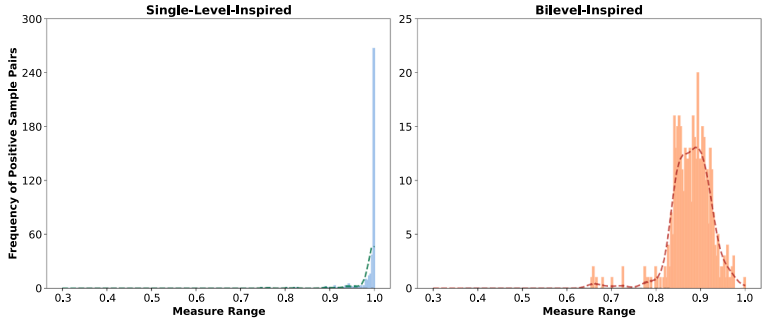
Fig. 3. Ablation studies of the proposed modules (*e.g.* INT-Xent and alignment) on all used datasets.



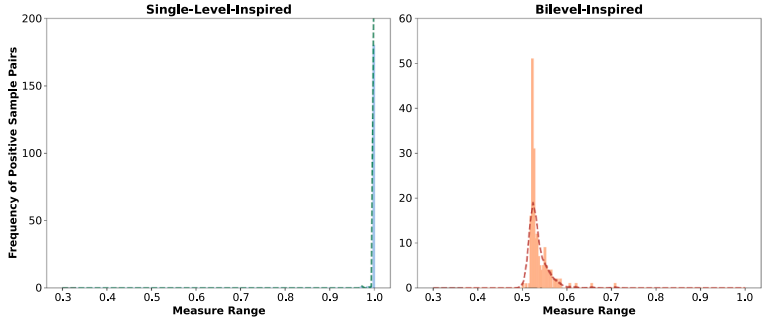
(a) ACM



(b) Squirrel

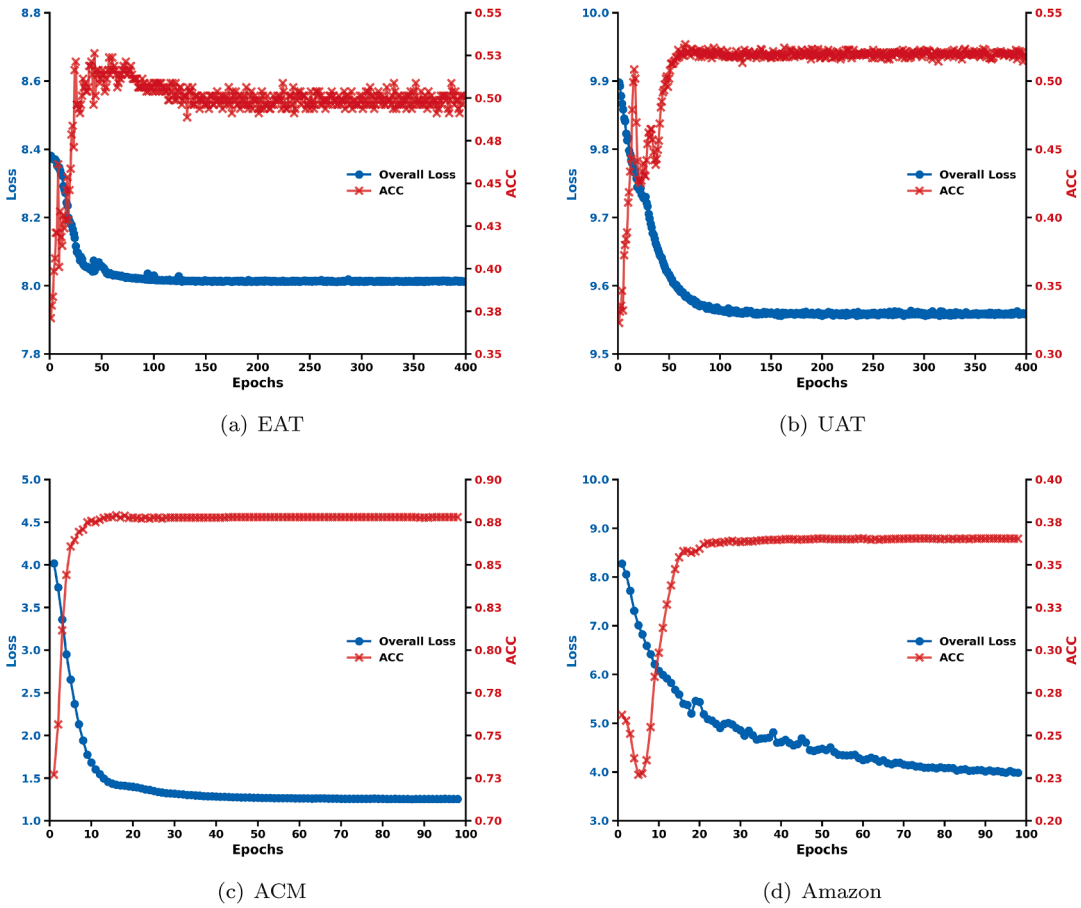


(c) EAT



(d) Cornell

Fig. 4. Histograms of two measures w.r.t similarity of positive pairs of embedded points. Our method outputs a more decentralized distribution (orange) compared to the classical one (blue). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



(a) EAT (b) UAT

(c) ACM (d) Amazon

Fig. 5. Convergence and accuracy performance curves.

4.7. A further look at hard sample pairs

In the previous section, we have demonstrated that INT-Xent has the ability to mine hard samples from the gradient perspective. Here, we will further deepen our understanding of its behavior through visualization. As shown in Fig. 4, NT-Xent, which relies only on instance-level similarity, is unable to measure more subtle differences between sample pairs, resulting in a long-tailed distribution (blue). In contrast, INT-Xent, enhanced with cluster-level knowledge, outputs a distribution closer to a normal curve (orange). Concretely, many of the samples of ACM considered very simple ($0.95 \sim 1$ in blue) by the NT-Xent are further diluted to varying degrees ($0.78 \sim 0.88$ in orange). Hard samples overlooked by NT-Xent ($0.90 \sim 0.95$ in blue) are effectively identified and shifted leftward by INT-Xent ($0.70 \sim 0.78$ in orange), leading to a more balanced concentration on the left half of the bell curve. In brief, this novel refined measure can be used to carve out finer hard samples and directly act on the gradient updating process.

4.8. Convergence analysis

As stated that the unified HALO works well from the gradient perspective, we will empirically study the convergence in the following. From the Fig. 5, the overall loss converges after around first 20% epochs and has smaller performance oscillations within rest of 80% epochs.

Our method not only converges quickly but also maintains consistent performance. One of reasons is that clustering consistency is guaranteed by a theorem.

4.9. The parameter analysis

For the customized measurement factor, we set a strong condition value $\xi = 2$ into Eq. (11). The reason is that $\xi = 2$ ensures $\left(\sum_{k=1}^{C} |p_{ik} - p_{i'k}| + 2 - 2S'_{ii}\right) \geq 0$ in Eq. (B.1). Different benchmarks have various values of $\sum_{k=1}^{C} |p_{ik} - p_{i'k}|$ and S'_{ii}, which means there is a certain smaller constant satisfied the non-negativity. We plot the line graphs of different values ($\xi = 0, 1, 2, 5, 10$) in Fig. 6.

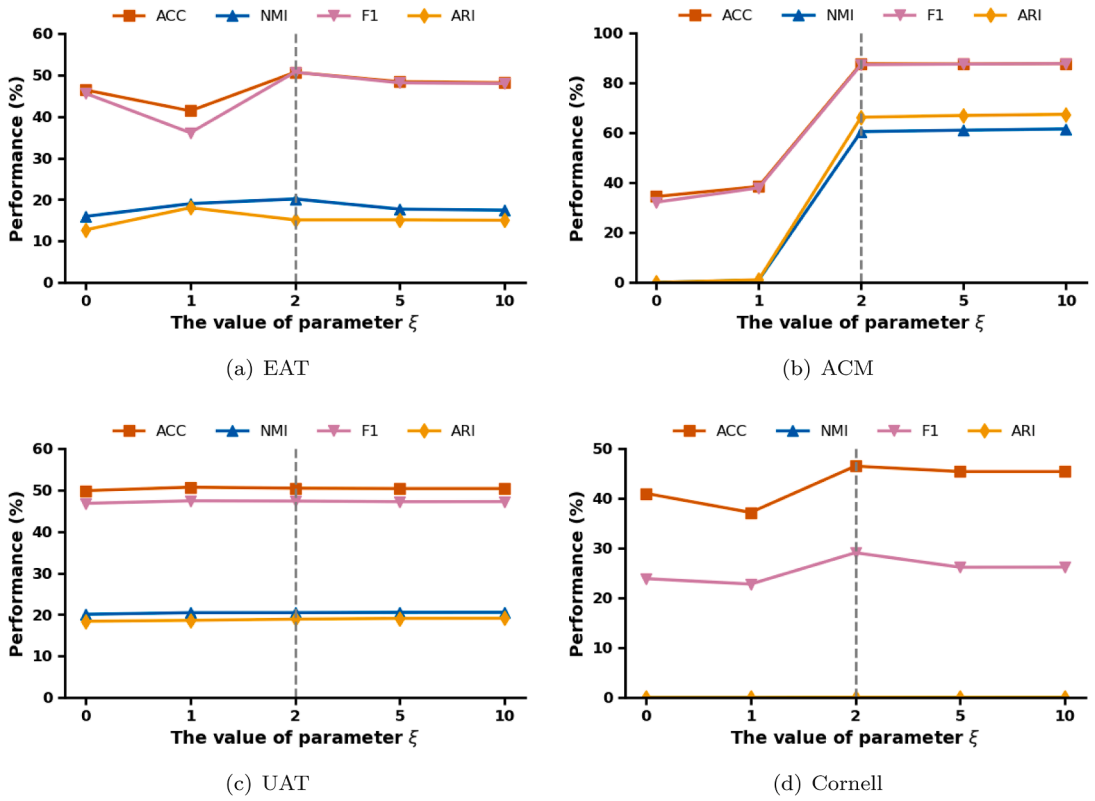


Fig. 6. Analysis of the hardness constant ξ on different datasets.

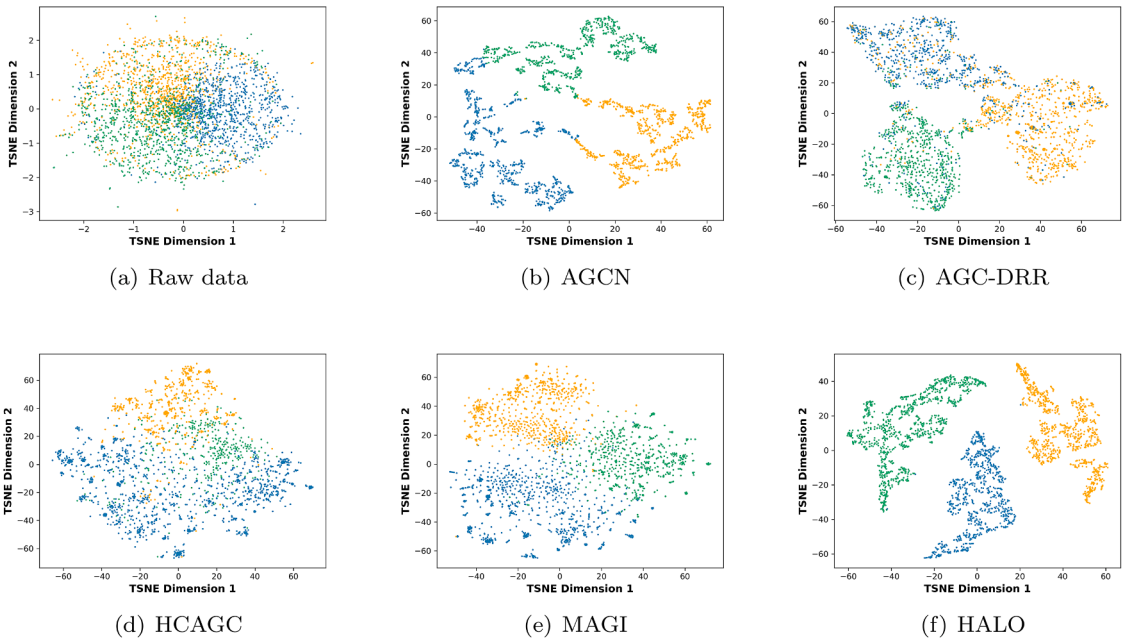
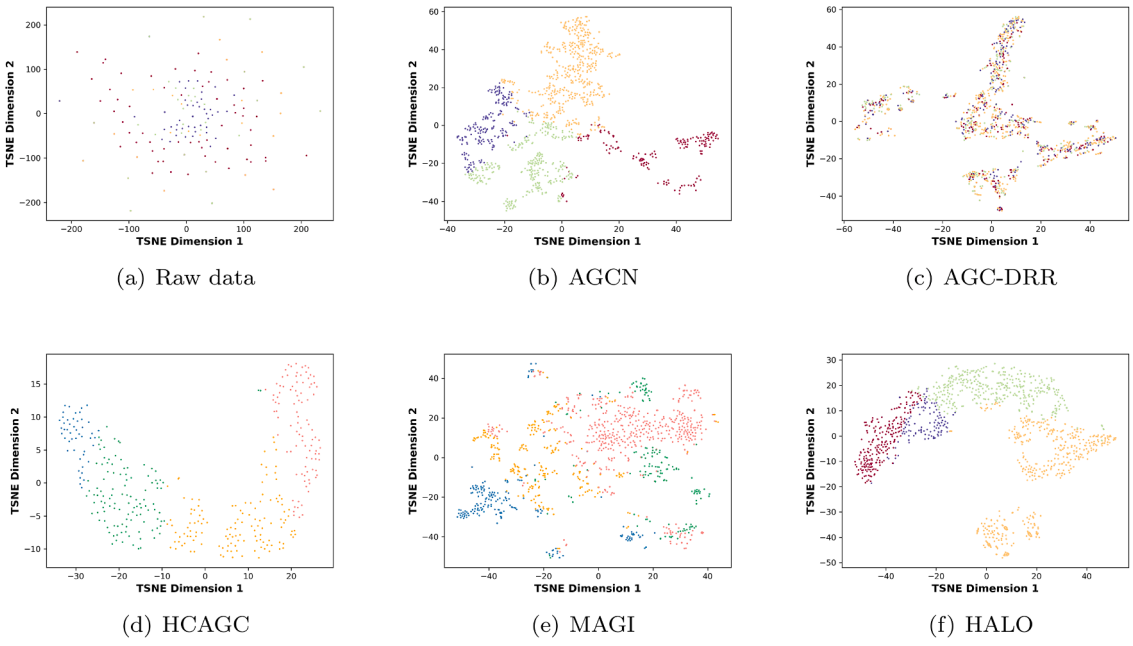


Fig. 7. Visualization of t-SNE on the simple dataset ACM.



(a) Raw data (b) AGCN (c) AGC-DRR

(d) HCAGC (e) MAGI (f) HALO

Fig. 8. Visualization of t-SNE on the overlapping dataset UAT.

One main conclusion is drawn: The parameter ξ smaller than the strong condition has a large effect on performance, while it has a small effect when larger than the strong condition. Since the strong condition already ensures the non-negativity, a larger constant only affects the normalized magnitude of the two measure levels. However, $\xi < 2$ can not guarantee the non-negativity and may inhibit performance, such as changes in Fig. 6(b). Therefore, in order to ensure the effect of mining hard positive samples while avoiding the dilution of the hardness measure, we finally choose the strong condition $\xi = 2$. We also analyze the effect of the trade-off parameter λ and teleport probability α. The results are shown in Appendix G.

4.10. The t-SNE visualization

In order to provide more insights of the clustering effect, t-SNE visualizations are conducted on two benchmarks as shown in Figs. 7 and 8. For the ACM dataset that has a few overlapping instances, all algorithms output obvious divisions generally. In specific, HALO, boosted by the hard samples mining technique, is able to produce the most clear structure with larger gap among clusters (see Fig. 7(f)). As for a more challenging dataset UAT that has numerous overlapping instances (see Fig. 8(a)), some hard samples still retain in various methods' results. Even worse, AGC-DRR does not distinguish between different classes of samples at all (see Fig. 8(c)). In contrast, our method reveals a more discriminative clustering pattern. To sum up, the visualizations show that our model improves the separation and compactness of the clusters with help of the bilevel measure.

5. Conclusion

This work tackles a key bottleneck in attributed graph clustering: the representation-uncertainty caused by structural noise and attribute conflicts. We proposed HALO, a bilevel-inspired contrastive clustering framework that integrates two structural cluster-level signals with instance-level similarity to form an uncertainty-aware hardness measure. This formulation identifies samples whose cross-level inconsistencies reveal underlying ambiguity, and adaptively strengthens their influence during optimization. Theoretical analysis shows that HALO naturally emphasizes uncertain regions in the embedding space, while an optimal-transport-based alignment maintains consistent clustering distributions and avoids degeneration. Experiments across diverse datasets confirm that HALO uncovers more informative hard pairs, generalizes reliably, and delivers state-of-the-art performance. Looking ahead, we aim to unify the representation and decision uncertainty together for further strengthening generalization and performance in deep clustering.

Data availability

Data will be made available on request.

CRediT authorship contribution statement

Yuchen Zhu: Writing – original draft, Visualization, Validation, Software, Formal analysis, Data curation, Conceptualization; **Kuang Zhou:** Writing – review & editing, Project administration, Methodology, Investigation, Funding acquisition; **Haishan Ye:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Conceptualization; **Guang Dai:** Software, Resources, Project administration, Data curation; **Ivor W. Tsang:** Writing – review & editing, Methodology, Conceptualization.

Declaration of competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the National Social Science Fund of China (No. 23BTJ053), the Natural Science Basic Research Plan in Shaanxi Province of China (No. 2025JCYBMS-678) and the National Natural Science Foundation of China (No. 92371101).

Appendix A. The Proof of Proposition 1

Proof. For positive pairs, we have

$$\frac{\partial \mathcal{L}(\boldsymbol{S})}{\partial S'_{ii}} = -\frac{1}{N}\frac{\partial\left[\sum_{i=1}^{N}[S'_{ii}/\tau - \log\sum_{j=1,j\neq i}^{N} e^{S'_{ij}/\tau}]\right]}{\partial S'_{ii}} = -\frac{1}{N\tau} < 0. \tag{A.1}$$

For negative pairs,

$$\begin{aligned}\frac{\partial \mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}} &= -\frac{1}{N}\frac{\partial\left[\sum_{i=1}^{N}[S'_{ii}/\tau - \log\sum_{j=1,j\neq i}^{N} e^{S'_{ij}/\tau}]\right]}{\partial S'_{ij}} \\ &= \frac{1}{N\tau}\frac{e^{S'_{ij}/\tau}}{\sum_{j=1,j\neq i}^{N} e^{S'_{ij}/\tau}} > 0.\end{aligned} \tag{A.2}$$

Eqs. (A.1) and (A.2) indicate that the theoretical gradient satisfies assumptions of the GPW $\left(i.e., \frac{\partial \mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}} \geq 0 \text{ and } \frac{\partial \mathcal{L}(\boldsymbol{S})}{\partial S'_{ii}} \leq 0\right)$. It is noted that $\frac{\partial \mathcal{L}(\boldsymbol{S})}{\partial S'_{ij}}$ increases with the increase of S'_{ij} because of $\frac{\partial^2 \mathcal{L}(\boldsymbol{S})}{\partial {S'_{ij}}^2} \geq 0$. To sum up, for negative pairs $(\boldsymbol{z_i}, \boldsymbol{z_j}')$, $\mathcal{L}$ can increase the weight w_{ij} according to a single sample pair during the learning process. By contrast, as for arbitrary positive pairs $(\boldsymbol{z_i}, \boldsymbol{z_i}')$, the weight $w_{ii} = 1/N\tau$ is a constant scalar. In other words, the loss function can not discriminate hard and easy positive sample pairs. □

Appendix B. The Proof of Proposition 2

Proof. For each positive sample pair,

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{int}}(\boldsymbol{S})}{\partial S'_{ii}} &= -\frac{1}{N}\left[\frac{1}{\|\boldsymbol{\alpha}'_{ii}\|\tau}\frac{\partial}{\partial S'_{ii}}\left(\sum_{k=1}^{C}\left|p_{ik} - p_{i'k}\right| - S'_{ii} + 2\right)S'_{ii}\right] \\ &= -\frac{1}{N\|\boldsymbol{\alpha}'_{ii}\|\tau}\left(\sum_{k=1}^{C}\left|p_{ik} - p_{i'k}\right| + 2 - 2S'_{ii}\right) \leq 0.\end{aligned} \tag{B.1}$$

Hence, $w_{ii} = \left|\frac{\partial \mathcal{L}_{\text{int}}(\boldsymbol{S})}{\partial S'_{ii}}\right| \nearrow (1 - S'_{ii})$. This draws the conclusion that the weight of positive sample pairs gets larger when these two samples are more disparate, and vice versa.

For each negative sample pair,

$$\begin{aligned}
\frac{\partial \mathcal{L}_{\text{int}}(\boldsymbol{S})}{\partial S'_{ij}} &= -\frac{1}{N}\frac{\partial}{\partial S'_{ij}}\left[\sum_{i=1}^{N}\tilde{\alpha}_{ii}S'_{ii}/\tau - \log\left(\sum_{j=1,j\neq i}^{N} e^{\tilde{\alpha}'_{ij}S'_{ij}/\tau}\right)\right] \\
&= \frac{1}{N}\frac{\partial}{\partial S'_{ij}}\left[\log\left(\sum_{j=1,j\neq i}^{N} e^{\tilde{\alpha}'_{ij}S'_{ij}/\tau}\right)\right] \\
&= \frac{1}{N}\frac{1}{\sum_{j=1,j\neq i}^{N} e^{\tilde{\alpha}'_{ij}S'_{ij}/\tau}}\frac{\partial}{\partial S'_{ij}}\left(e^{\tilde{\alpha}'_{ij}S'_{ij}/\tau}\right) \\
&= \frac{1}{N\tau\|\alpha'_{ij}\|}\underbrace{\frac{1}{\sum_{j=1,j\neq i}^{N} e^{\tilde{\alpha}'_{ij}S'_{ij}/\tau}} e^{\tilde{\alpha}'_{ij}S'_{ij}/\tau}\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+2S'_{ij}\right)}_{=m(S'_{ij})} \\
&= \frac{1}{N\tau\|\alpha'_{ij}\|}m(S'_{ij}) \geq 0.
\end{aligned}$$

If we want to get a general idea of the correlation that the more similar negative pairs are, the larger weight of them, we just need to verify $\frac{m(S'_{ij}+\epsilon)}{m(S'_{ij})} > 1, S'_{ij} < \epsilon + S'_{ij} \leq 1$. In fact,

$$\begin{aligned}
\frac{m(S'_{ij}+\epsilon)}{m(S'_{ij})} &= \frac{\frac{e^{\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+\epsilon+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}{c_1+e^{\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+\epsilon+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+2\epsilon+2S'_{ij}\right)}{\frac{e^{\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}{c_1+e^{\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}\left(\sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2+2S'_{ij}\right)} \\
&= \frac{\frac{e^{\left(c_2+\epsilon+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}{c_1+e^{\left(c_2+\epsilon+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}\left(c_2+2\epsilon+2S'_{ij}\right)}{\frac{e^{\left(c_2+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}{c_1+e^{\left(c_2+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}\left(c_2+2S'_{ij}\right)} \\
&= \frac{c_1+e^{\left(c_2+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}{c_1+e^{\left(c_2+\epsilon+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}\frac{e^{\left(c_2+\epsilon+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}{e^{\left(c_2+S'_{ij}\right)S'_{ij}/\tilde{\tau}}}\left(1+\frac{2\epsilon}{c_2+2S'_{ij}}\right) \\
&> e^{-\epsilon S'_{ij}/\tilde{\tau}}e^{\epsilon S'_{ij}/\tilde{\tau}}\left(1+\frac{2\epsilon}{c_2+2S'_{ij}}\right) > 1,
\end{aligned}$$

where $\tilde{\tau} = \tau\|\alpha'_{ij}\|$, c_1 is a summation over non-i,j and $c_2 \triangleq \sum_{k=1}^{C}\left|p_{ik}-p_{j'k}\right|+2$. Therefore, $w_{ij} = \left|\frac{\partial \mathcal{L}_{\text{int}}(\boldsymbol{S})}{\partial S'_{ij}}\right| \nearrow S'_{ij}$. □

Appendix C. Algorithm

Algorithm 2 Fast Sinkhorn Algorithm.

Input: Probabilistic assignment matrix $\mathbf{P}$, regularization parameter $1/\eta$, iteration count sk.
Output: Probabilistic assignment matrix $\mathbf{P}^*$.

$\mathbf{Q} \leftarrow \exp(\eta\mathbf{P})$.
Normalize: $\mathbf{Q} \leftarrow \mathbf{Q}/\sum_{i,j}\mathbf{Q}_{ij}$.
for $t = 1$ to sk **do**
  Normalize rows: $\mathbf{Q}_{i,:} \leftarrow \mathbf{Q}_{i,:}/\sum_j \mathbf{Q}_{i,j}$ for all i.
  Scale rows: $\mathbf{Q} \leftarrow \mathbf{Q}/C$.
  Normalize columns: $\mathbf{Q}_{:,j} \leftarrow \mathbf{Q}_{:,j}/\sum_i \mathbf{Q}_{i,j}$ for all j.
  Scale columns: $\mathbf{Q} \leftarrow \mathbf{Q}/N$.
end for
Scale back: $\mathbf{P}^* \leftarrow \mathbf{Q} \cdot N$.

Appendix D. The Proof of Theorem 1

Proof. Suppose ϵ_1 and ϵ_2 are the optimization error of $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively. We have

$$0 \leq \mathcal{L}_1 = -\sum_{i=1}^{N}\sum_{k=1}^{C} p^*_{ik} \log p^*_{i'k} \leq \epsilon_1, \tag{D.1}$$

$$0 \leq \mathcal{L}_2 = -\sum_{i=1}^{N}\sum_{k=1}^{C} p^*_{i'k} \log p^*_{ik} \leq \epsilon_2. \tag{D.2}$$

The following inequality holds by subtracting Eq. (D.1) from Eq. (D.2),

$$-\epsilon_1 \leq -\sum_{i=1}^{N}\sum_{k=1}^{C} \left[p^*_{i'k} \log p^*_{ik} + \left(p^*_{i'k} \log p^*_{i'k} - p^*_{i'k} \log p^*_{i'k}\right) - p^*_{ik} \log p^*_{i'k}\right] \leq \epsilon_2. \tag{D.3}$$

Then

$$-\epsilon_1 \leq -\sum_{i=1}^{N}\sum_{k=1}^{C} \left[\left(p^*_{i'k} - p^*_{ik}\right) \log p^*_{i'k} + p^*_{i'k} \log \frac{p^*_{ik}}{p^*_{i'k}}\right] \leq \epsilon_2. \tag{D.4}$$

With further minimization, $\epsilon_1 \to 0$ and $\epsilon_2 \to 0$. That is,

$$\sum_{i=1}^{N}\sum_{k=1}^{C} \left[\left(p^*_{i'k} - p^*_{ik}\right) \log p^*_{i'k} + p^*_{i'k} \log \frac{p^*_{ik}}{p^*_{i'k}}\right] \to 0. \tag{D.5}$$

Through a case-by-case analysis of the cases based on the relative values of p^*_{ik} and $p^*_{i'k}$, it can be concluded that Eq. (D.5) leads to $\mathbf{P'}^* \to \mathbf{P}^*$.

□

Appendix E. Empirical Runtime and Memory Analysis

To empirically validate the scalability of HALO, we conducted a comprehensive evaluation of runtime and GPU memory consumption across three datasets of varying scales. The results are detailed in Tables E.3–E.5. For fairness, baselines such as Dink-Net, MAGI, and FDAGC are excluded as their official GPU-accelerated implementations are unavailable. The reported runtime measures the training phase, excluding data loading and logging overheads.

(1) Memory Efficiency. The experimental results demonstrate that HALO strikes an optimal balance between effectiveness and resource consumption. Notably, HALO exhibits superior memory efficiency compared to contrastive learning counterparts. For instance, compared to the memory-intensive HCAGC, our method achieves an average of 5× reduction in peak memory usage. On the larger Amazon dataset (Table E.5), while methods like MVGRL fail due to Out-Of-Memory (OOM) errors, HALO maintains a manageable memory footprint, proving its capability to handle large-scale graphs.

(2) Computational Time. Regarding runtime, although HALO, MVGRL, and AGC-DRR all employ Siamese-like architectures, HALO benefits significantly from its prototype-based representation. Unlike MVGRL, which suffers from high complexity $\mathcal{O}(d^2N + N^2d)$ leading to prohibitive costs on large graphs, HALO optimizes the process with a reduced complexity of $\mathcal{O}(C^2N + N^2d)$. Empirically, HALO saves approximately 75% of the runtime on average compared to AGC-DRR. While simpler methods like AGCN run faster, they lack the uncertainty-modeling capability; HALO thus provides a competitive trade-off, offering robust performance without the excessive computational burden seen in other advanced baselines. In summary, boosted by its unified framework, HALO is both computationally efficient and effective.

Table E.3
Comparison of peak GPU memory usage and training time on the **Squirrel** dataset. Note that for clarity, the **highest costs** (worst performance) are highlighted in **bold**, and the runners-up are underlined, to emphasize the efficiency bottlenecks of baselines.

Method	AGCN	MVGRL	AGC-DRR	ProGCL	HCAGC	HCHSM	HALO
Running Time (s)	27.29	499.42	**2212.44**	242.98	121.41	80.38	378.40
Max Memory (MB)	1355.73	**22577.68**	5224.54	6286.46	11112.95	6064.22	3703.05

Table E.4
Comparison of peak GPU memory usage and training time on the **ACM** dataset.

Method	AGCN	MVGRL	AGC-DRR	ProGCL	HCAGC	HCHSM	HALO
Running Time (s)	20.69	200.66	**456.80**	117.30	91.03	51.24	150.79
Max Memory (MB)	261.64	**12931.09**	1113.75	686.90	6114.84	3494.21	616.08

Table E.5
Comparison of peak GPU memory usage and training time on the large-scale **Amazon** dataset. Note: "-" indicates failure to converge due to numerical instability.

Method	AGCN	MVGRL	AGC-DRR	ProGCL	HCAGC	HCHSM	HALO
Running Time (s)	535.38	**N/A**	52077.79	–	649.55	711.16	10403.29
Max Memory (MB)	20779.43	**OOM**	39831.26	34123.05	64734.71	40977.28	32062.00

Table E.6
Performance comparison of HALO using different backbones (Standard GCN vs. GraphSAGE) across six datasets. The best performance for each metric on each dataset is highlighted in **bold**.

Dataset	ACC		NMI		ARI		F1	
	GCN (Ours)	SAGE	GCN (Ours)	SAGE	GCN (Ours)	SAGE	GCN (Ours)	SAGE
ACM	**87.65**	60.03	**60.45**	24.23	**66.21**	26.42	**87.23**	57.84
EAT	**50.07**	41.10	**19.93**	10.22	**18.06**	7.46	**49.10**	40.00
UAT	**50.74**	50.00	**23.42**	17.34	**20.35**	16.00	47.32	**48.70**
Cornell	43.50	**53.55**	4.47	**5.87**	**5.39**	2.00	**25.08**	18.35
Amazon	35.17	**36.82**	**1.25**	0.05	0.00	**0.02**	**23.51**	10.91
Squirrel	25.87	**27.80**	1.91	**4.52**	1.41	**2.13**	**25.64**	23.19

Appendix F. Ablation Study of the Backbone

To investigate the generalization capability of the HALO framework and verify whether the performance gains are tied specifically to the GCN encoder, we conducted an ablation study by replacing the standard GCN backbone with GraphSAGE (using mean aggregation). We evaluated the performance across all six datasets, and the comparative results are presented in Table E.6. The results indicate that HALO is backbone-agnostic. The proposed bilevel optimization mechanism functions effectively with different encoders, consistently producing valid clustering results. This confirms that the core contribution–mitigating representation uncertainty–is a universal enhancement that does not strictly rely on the spectral properties of GCN.

In specific, we observe an interesting trade-off dictated by the graph topology and scale. On homophilous graphs (*e.g.*, ACM, UAT), the standard GCN encoder significantly outperforms GraphSAGE. This is expected in transductive clustering settings with high homophily, where the full-batch spectral convolution of GCN better preserves the global community structure compared to the neighbor sampling strategy of GraphSAGE. On heterophilous and large-scale graphs (*e.g.*, Cornell, Amazon), conversely, GraphSAGE demonstrates slightly superior performance on datasets with lower homophily or larger scales. Notably, on the Cornell dataset, switching to GraphSAGE boosts the accuracy from 43.50% to 53.55%. This suggests that the aggregation mechanism of GraphSAGE is more effective at filtering out noisy inter-class connections in heterophilous settings. On the large-scale Amazon and noise-sensitive Squirrel datasets, GraphSAGE also yields higher accuracy (36.82% and 27.80%, respectively), highlighting the scalability potential of the HALO framework when paired with inductive-friendly backbones.

While GCN is adopted as the default backbone for its superior performance on standard benchmarks, HALO offers the flexibility to switch to GraphSAGE to maximize performance on specific graph types like heterophilous networks, further validating the versatility of our framework.

Appendix G. Extra Sensitivity Analysis

The hyperparameter λ balances the representation learning loss ($\mathcal{L}_{int}$) and the cluster consistency loss ($\mathcal{L}_{clus}$). We evaluated λ across the magnitude range of $\{0.01, 0.1, 1, 5, 10\}$. The results are summarized in Fig. G.1. HALO exhibits remarkable stability with respect to λ. Basically, the performance variance is minimal within the range of $[0.1, 10]$, indicating that the model is not sensitive to the exact weight of this alignment term. The default setting $\lambda = 0.1$ provides the best average performance across datasets. Thus, we adopt $\lambda = 0.1$ as the universal default to ensure generalization. Additionally, this wide stable range confirms that the OT-based alignment ($\mathcal{L}_{clus}$) works synergistically with the contrastive learning ($\mathcal{L}_{int}$).

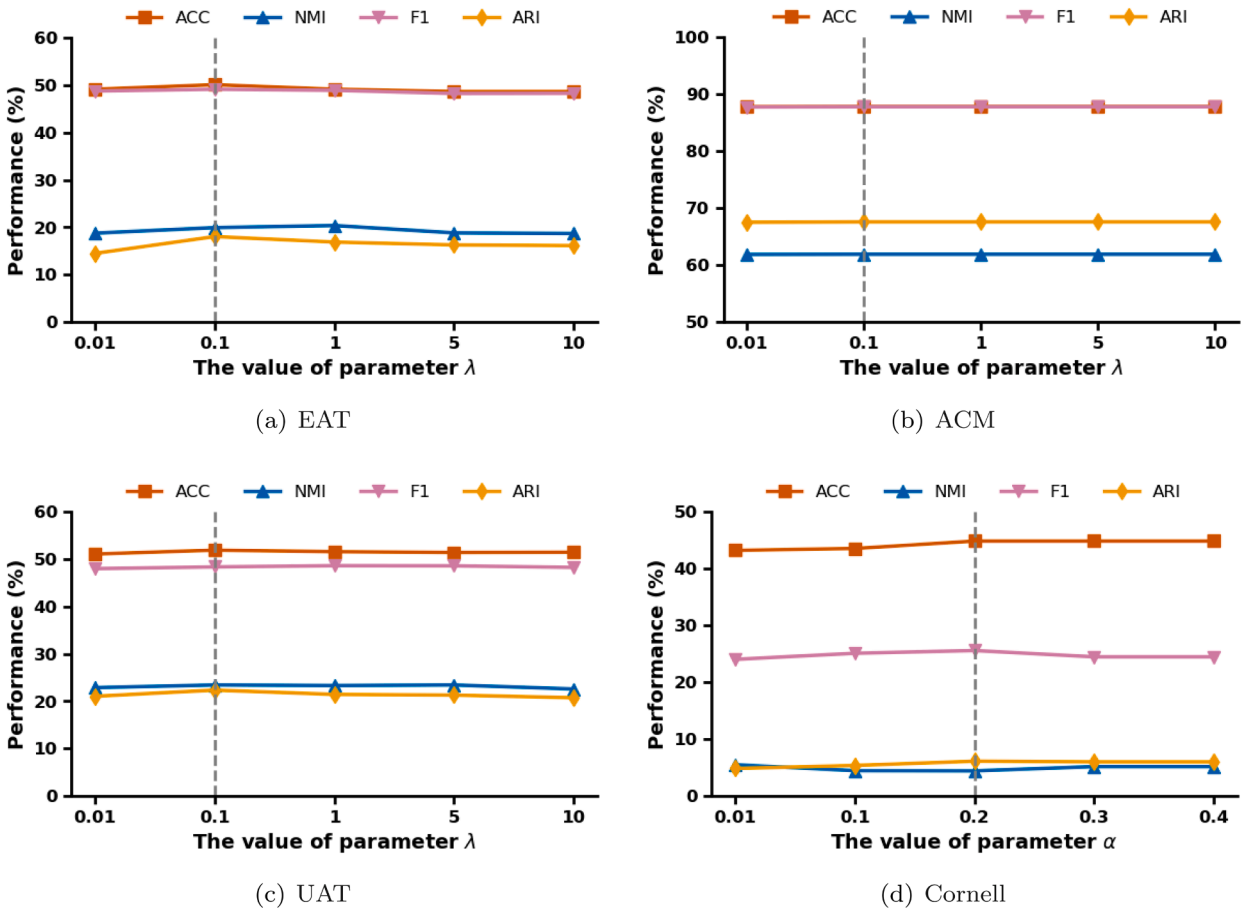


Fig. G.1. Analysis of the trade-off parameter λ on different datasets.

To understand the impact of the diffusion mechanism used in HALO, we conducted a detailed sensitivity analysis on the PPR diffusion parameter α. We varied α within the range of $\{0.01, 0.1, 0.2, 0.3, 0.4\}$. The performance trends across four datasets are presented in Fig. G.2. The parameter α governs the trade-off between structural locality and global context exploration. A smaller α facilitates longer random walks to capture higher-order proximity, whereas a larger α confines the diffusion to the local neighborhood. We observe the following key patterns: (1) General Trend: As observed in datasets EAT, UAT, and ACM, the performance consistently follows an inverted-U pattern, peaking at $\alpha \in [0.1, 0.2]$. For instance, on EAT, the accuracy rises from around 38% ($\alpha = 0.01$) to a peak of around 50% ($\alpha = 0.2$) before dropping to around 41% ($\alpha = 0.4$). This confirms that a moderate expansion of the receptive field is crucial. It introduces global structural signals that are complementary to the local adjacency view, maximizing the efficacy of cross-view contrastive learning. However, as α increases beyond 0.3, the diffusion view becomes overly localized and structurally redundant to the adjacency view, leading to a performance drop. (2) Topology-Dependent Preference: The Cornell dataset exhibits a distinct trajectory, where performance improves monotonically as α increases, achieving a high of around 50% at $\alpha = 0.4$. As shown in the Table 1, Cornell is known for both low homophily and sparse structure ($\bar{d} = 1.89$). In such topologies, long-range random walks (induced by small α) may inadvertently aggregate noisy information from nodes of different classes. Therefore, a larger α (stronger teleport probability) acts as a necessary regularizer, constraining the diffusion to a reliable local neighborhood and filtering out long-range noise. This observation highlights HALO's adaptability to varying structural properties. Based on these results, we adopt $\alpha = 0.2$ as the robust default for all datasets to balance global exploration and local preservation. Meanwhile, for heterophilic datasets like Cornell, a larger α is recommended to ensure structural reliability.

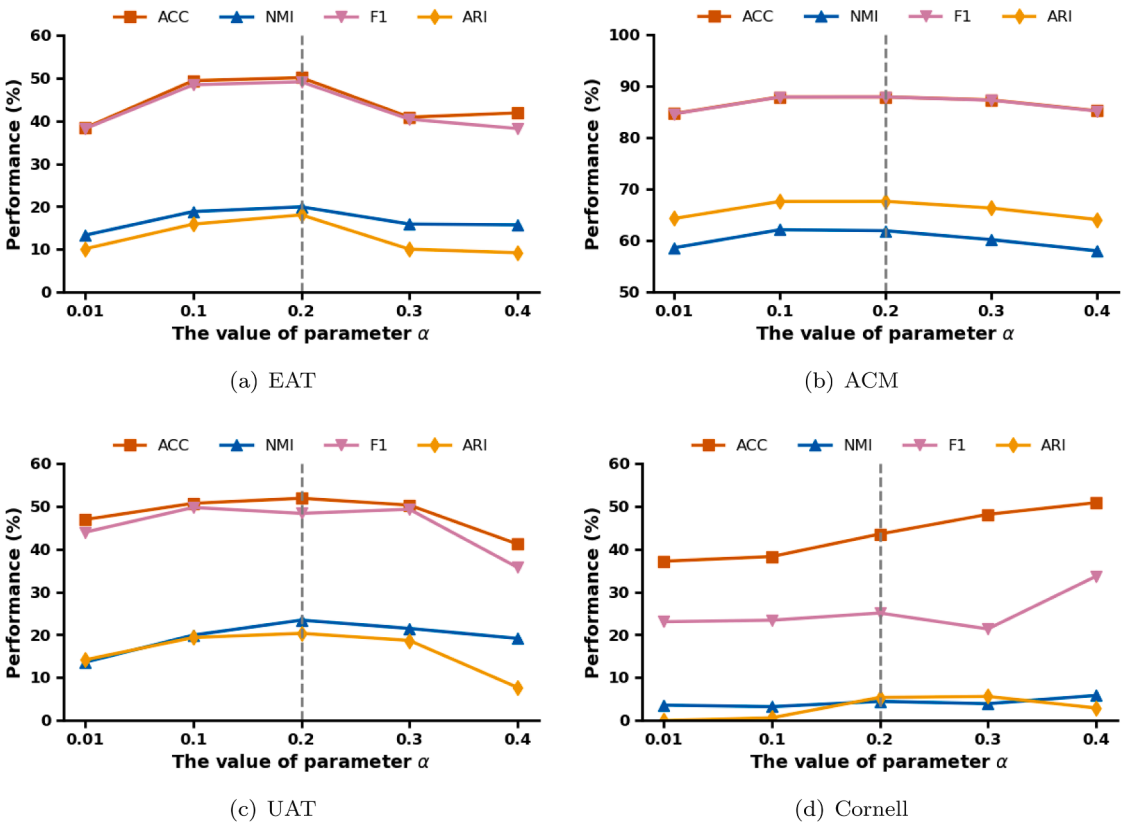


Fig. G.2. Analysis of the teleport probability parameter α of the graph diffusion on different datasets.

References

[1] E. Yan, S. Hao, T. Zhang, T. Hao, Q. Chen, J. Yu, Graph representation learning method based on three-way partial order structure, Int. J. Approximate Reasoning 165 (2024) 109104.
[2] Z. Kang, X. Xie, B. Li, E. Pan, CDC: a simple framework for complex data clustering, IEEE Trans. Neural Netw. Learn. Syst. 36 (7) (2025) 13177–13188.
[3] Z. Fang, S. Du, Y. Zou, Y. Tan, N. Song, S. Wang, Be reliable: an interpretable attribute-oriented representation learning framework, IEEE Trans. Neural Netw. Learn. Syst. 37 (3) (2025) 1134-1148.
[4] X. Yang, Y. Wang, J. Chen, W. Fan, X. Zhao, E. Zhu, X. Liu, D. Lian, Dual test-time training for out-of-distribution recommender system, IEEE Trans. Knowl. Data Eng. 37 (6) (2025) 3312–3326.
[5] Y. Mo, H.T. Shen, X. Zhu, Efficient self-supervised heterogeneous graph representation learning with reconstruction, Inf. Fusion 117 (2025) 102846.
[6] J. Robinson, C.-Y. Chuang, S. Sra, S. Jegelka, Contrastive learning with hard negative samples, in: International Conference on Learning Representations, 2021.
[7] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: International Conference on Machine Learning, PMLR, 2020, pp. 1597–1607.
[8] Y. Zheng, C. Jia, J. Yu, Attributed graph clustering under the contrastive mechanism with cluster-preserving augmentation, Inf. Sci. 681 (2024) 121225.
[9] Z. Chen, L. Li, X. Zhang, H. Wang, Deep graph clustering via aligning representation learning, Neural Netw. 183 (2025) 106927.
[10] K. Gao, M. Chen, C. Liu, S. Xue, Z. Qiu, T. Ren, X. Jia, W. Hu, A debiased graph clustering approach using dual contrastive learning, in: 2024 IEEE International Conference on Web Services, 2024, pp. 1198–1205.
[11] W. Tu, S. Zhou, X. Liu, C. Ge, Z. Cai, Y. Liu, Hierarchically contrastive hard sample mining for graph self-supervised pretraining, IEEE Trans. Neural Netw. Learn. Syst. 35 (11) (2024) 16748–16761.
[12] J. Xia, L. Wu, G. Wang, J. Chen, S.Z. Li, ProGCL: rethinking hard negative mining in graph contrastive learning, in: International Conference on Machine Learning, PMLR, 2022, pp. 24332–24346.
[13] S. Zhou, H. Xu, Z. Zheng, J. Chen, Z. Li, J. Bu, J. Wu, X. Wang, W. Zhu, M. Ester, A comprehensive survey on deep clustering: taxonomy, challenges, and future directions, ACM Comput. Surv. 57 (3) (2024) 1–38.
[14] Y. Zhang, Y. Yuan, Q. Wang, Multi-level graph contrastive prototypical clustering, in: IJCAI, 2023, pp. 4611–4619.
[15] L. Gong, S. Zhou, W. Tu, X. Liu, Attributed graph clustering with dual redundancy reduction, in: IJCAI, 2022, pp. 3015–3021.
[16] Y. Li, P. Hu, Z. Liu, D. Peng, J.T. Zhou, X. Peng, Contrastive clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, 35, 2021, pp. 8547–8555.
[17] Z. Ma, Á. López-Oriona, H. Ombao, Y. Sun, FCPCA: fuzzy clustering of high-dimensional time series based on common principal component analysis, Int. J. Approximate Reasoning 187 (2025) 109552.
[18] H. Yang, F. Yu, W. Pedrycz, Z. Yang, J. Chang, J. Wang, An auto-weighted enhanced horizontal collaborative fuzzy clustering algorithm with knowledge adaption mechanism, Int. J. Approximate Reasoning 169 (2024) 109169.
[19] L. Guiziou, E. Ramasso, S. Thibaud, S. Denneulin, DEEM: a novel approach to semi-supervised and unsupervised image clustering under uncertainty using belief functions and convolutional neural networks, Int. J. Approximate Reasoning 181 (2025) 109400.
[20] Y. Zhu, K. Zhou, F. Cuzzolin, TDCC: A Trustworthy Deep Credal Clustering Method for Uncertain Data, IEEE Trans. Cybern. (2026).
[21] M. Cai, Z. Wu, Q. Li, F. Xu, J. Zhou, GFDC: a granule fusion density-based clustering with evidential reasoning, Int. J. Approximate Reasoning 164 (2024) 109075.
[22] K. Zhou, Y. Zhu, M. Guo, M. Jiang, MvWECM: multi-view weighted evidential c-means clustering, Pattern Recognit. 159 (2025) 111108.

[23] R. Zhang, H. Zhang, Y. Qian, Three-way space structure and clustering of categorical data, Int. J. Approximate Reasoning 184 (2025) 109457.
[24] K. Wang, F. Cuzzolin, K. Shariatmadar, D. Moens, H. Hallez, A review of uncertainty representation and quantification in neural networks, IEEE Trans. Pattern Anal. Mach. Intell. 48 (2026) 2476-2495.
[25] K. Zhou, Z. Zhang, H. Xu, L. Wang, Y. Shi, Reliability Analysis Based on Evidential Likelihood for Uncertain Mixed Weibull Distribution, IEEE Trans. Reliab. 75 (2026) 1020–1034.
[26] L. Chang, X. Niu, Z. Li, Z. Zhang, S. Li, P. Fournier-Viger, ULDC: uncertainty-based learning for deep clustering, Appl. Intell. 55 (3) (2025) 223.
[27] L. Chang, L. Chen, C. Zhou, Uncertainty-aware contrastive learning for deep clustering, Neurocomputing 647 (2025) 130568 .
[28] H. Zhao, X. Yang, Z. Wang, E. Yang, C. Deng, Graph debiased contrastive learning with joint representation clustering, in: IJCAI, 2021, pp. 3434–3440.
[29] L. Zhu, H. Sun, X. Huang, P. Lou, L. He, Contrastive deep graph clustering with hard boundary sample awareness, Inf. Process. Manage. 62 (3) (2025) 104050.
[30] Y. Liu, W. Tu, S. Zhou, X. Liu, L. Song, X. Yang, E. Zhu, Deep graph clustering via dual correlation reduction, in: Proceedings of the AAAI Conference on Artificial Intelligence, 36, 2022, pp. 7603–7611.
[31] S. Yi, W. Ju, Y. Qin, X. Luo, L. Liu, Y. Zhou, M. Zhang, Redundancy-free self-supervised relational learning for graph clustering, IEEE Trans. Neural Netw. Learn. Syst. 35 (12) (2024) 18313–18327.
[32] L. Page, The PageRank Citation Ranking: Bringing Order to the Web, Technical Report, 1999.
[33] X. Chen, K. He, Exploring simple siamese representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15750–15758.
[34] J. Cai, Y. Zhang, J. Fan, Y. Du, W. Guo, Dual contrastive graph-level clustering with multiple cluster perspectives alignment, in: IJCAI, 2024.
[35] X. Yan, Y. Mao, M. Li, Y. Ye, H. Yu, Multitask image clustering via deep information bottleneck, IEEE Trans. Cybern. 54 (3) (2024) 1868–1881.
[36] X. Wang, X. Han, W. Huang, D. Dong, M.R. Scott, Multi-similarity loss with general pair weighting for deep metric learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5022–5030.
[37] R. Li, S. Di, L. Chen, X. Zhou, GradGCL: gradient graph contrastive learning, in: International Conference on Data Engineering, 2024, pp. 1171–1184.
[38] C.-H. Yeh, C.-Y. Hong, Y.-C. Hsu, T.-L. Liu, Y. Chen, Y. LeCun, Decoupled contrastive learning, in: European Conference on Computer Vision, Springer, 2022, pp. 668–684.
[39] M. Cuturi, Sinkhorn distances: lightspeed computation of optimal transport, in: Advances in Neural Information Processing Systems, 26, 2013, p. 2292-2300.
[40] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the International Conference on Artificial Intelligence and Statistics, 2010, pp. 249–256.
[41] Y. Liu, X. Yang, S. Zhou, X. Liu, Z. Wang, K. Liang, W. Tu, L. Li, J. Duan, C. Chen, Hard sample aware network for contrastive deep graph clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, 37, 2023, pp. 8914–8922.
[42] E. Pan, Z. Kang, Beyond homophily: reconstructing structure for graph-agnostic clustering, in: International Conference on Machine Learning, PMLR, 2023, pp. 26868–26877.
[43] O. Platonov, D. Kuznedelev, M. Diskin, A. Babenko, L. Prokhorenkova, A critical look at evaluation of GNNs under heterophily: are we really making progress?, in: International Conference on Learning Representations, 2023.
[44] L. Lovász, M.D. Plummer, Matching Theory, 367, American Mathematical Soc., 2009.
[45] K. Hassani, A.H. Khasahmadi, Contrastive multi-view representation learning on graphs, in: International Conference on Machine Learning, 2020, pp. 3451–3461.
[46] Z. Peng, H. Liu, Y. Jia, J. Hou, Attention-driven graph clustering network, in: ACM Multimedia, 2021, pp. 935–943.
[47] Y. Liu, K. Liang, J. Xia, S. Zhou, X. Yang, X. Liu, S.Z. Li, Dink-net: neural clustering on large graphs, in: International Conference on Machine Learning, PMLR, 2023, pp. 21794–21812.
[48] Y. Yang, X. Su, B. Zhao, G. Li, P. Hu, J. Zhang, L. Hu, Fuzzy-based deep attributed graph clustering, IEEE Trans. Fuzzy Syst. 32 (4) (2024) 1951–1964.
[49] Y. Liu, J. Li, Y. Chen, R. Wu, E. Wang, J. Zhou, S. Tian, S. Shen, X. Fu, C. Meng, et al., Revisiting modularity maximization for graph clustering: a contrastive learning perspective, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, pp. 1968–1979.
[50] M.-S. Chen, P.-Y. Lai, D.-Z. Liao, C.-D. Wang, J.-H. Lai, Homophily induced contrastive attributed graph clustering, IEEE Trans. Circuits Syst. Video Technol. 35 (10) (2025) 10213–10224.