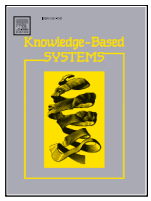
Knowledge-Based Systems 339 (2026) 115598

Contents lists available at ScienceDirect

Knowledge-Based Systems

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



HINMOIT: Representation learning on heterogeneous information networks with multi-order information interaction

Yanglan Gan [a], Lideng Cai [a], Kaili Wang [a], Guangwei Xu [a], Guobing Zou [b,*]

[a] School of Computer Science and Technology, Donghua University, Shanghai, 201620, China
[b] School of Computer Engineering and Science, Shanghai University, Shanghai, 201620, China

ARTICLE INFO

Keywords:
Heterogeneous information network
Multi-order information extraction
Gated interaction mechanism
Graph representation learning

ABSTRACT

Heterogeneous graph neural networks (HGNNs) have emerged as powerful tools for modeling complex graph-structured data. However, existing methods adapted from homogeneous graph neural networks either homogenize graphs via meta-paths or directly model relations, failing to capture multi-scale information and suffering from over-smoothing with multi-layer convolutions. The hybrid models also encounter feature space discrepancies during alignment and fusion. To address these challenges, we propose HINMOIT, a novel representation learning framework on heterogeneous information networks that integrates multi-order information extraction and interaction. HINMOIT consists of three key modules, including a low-order relation-aware aggregation encoder, a high-order meta-graph aggregation encoder and a multi-order information interaction module. The low-order relation-aware encoder captures local structure using residual connections to preserve node information across layers. In contrast, the high-order meta-graph encoder extracts high-level semantics through 1-hop aggregation. The interaction module learns consensus representations through progressive alignment and fusion of multi-order information between layers of the two encoders. Crucially, both encoders incorporate topology-aware aggregation coefficients derived from structural semantics to improve neighbor selection. Experiments on three benchmark datasets demonstrate that HINMOIT effectively captures rich structural information, significantly outperforming state-of-the-art methods on downstream tasks.

1. Introduction

Graph-structured data is prevalent across various domains, ranging from social networks [1] and recommendation systems [2] to bioinformatics [3] and knowledge graphs [4]. Particularly, heterogeneous information networks (HINs) [5,6] are valuable for modeling complex systems with multi-typed nodes and edges. To process these complex structures, graph neural networks (GNNs) [7,8] have emerged as powerful tools for graph representation learning. However, conventional GNNs designed for homogeneous graphs fail to effectively model heterogeneous relations [9–11], promoting the development of specialized heterogeneous GNNs (HGNNs) [12,13].

HGNN methods can be broadly classified into two categories, including meta-path-based [9,14] and relation-based methods [10,11], which respectively model node interactions through meta-path-induced views and type-specific message passing. Fundamentally, these two paradigms capture structural dependencies at distinct granularity. Relation-based encoders focus on short-range dependencies within immediate neighborhoods, whereas meta-path-based encoders extract long-range semantic correlations along predefined paths. Meta-path-based methods facilitate the homogenization of graphs via domain-specific paths, enabling the modeling of long-range semantics. However, they usually neglect the primitive local structures, despite the critical role of 1-hop neighborhoods in representation [15]. Differently, relation-based methods directly learn node representation on the heterogeneous graph through edge-type-aware aggregation. While preserving the original graph structure, they are limited to low-order neighborhood information and may overlook long-range dependencies critical for comprehensive node representation.

Recognizing the complementary strengths of low-order neighbor and high-order semantic modeling, recent hybrid HGNNs [16,17] attempt to combine a meta-path modeling paradigm that captures high-order semantics with a low-order modeling paradigm that preserves typed-edge message passing on the original HIN. However, such integration is non-trivial due to the inherent divergence in inductive biases and aggregation domains. High-order subgraph encoders aggregate along

* Corresponding author.
E-mail addresses: ylgan@dhu.edu.cn (Y. Gan), 2232844@mail.dhu.edu.cn (L. Cai), kailiwang@dhu.edu.cn (K. Wang), gwxu@dhu.edu.cn (G. Xu), gbzou@shu.edu.cn (G. Zou).

https://doi.org/10.1016/j.knosys.2026.115598
Received 27 August 2025; Received in revised form 10 January 2026; Accepted 20 February 2026
Available online 22 February 2026

path-induced contexts and emphasize composite semantics, whereas low-order subgraph encoders propagate local signals through ego neighborhoods. This fundamental discrepancy often yields semantic divergence in the latent space. Consequently, conventional late-stage fusion strategies, such as concatenation or static weighted summation, tend to heuristically merge non-equivalent features, leading to unstable optimization and compromised complementarity. For instance, representative pipelines like HMSG [16], which fuse meta-path subgraph embeddings, frequently suffer from semantic misalignment in the absence of explicit cross-view calibration. Furthermore, the efficacy of multi-view fusion has been scrutinized [18]. When distinct views encode genuinely heterogeneous semantics, fusing them without rigorous latent space alignment can result in semantic inconsistency, where the fused representations fail to encapsulate a coherent consensus across multiple views. Therefore, explicitly mitigating these feature-space discrepancies through dynamic interaction is essential for exploiting the synergy between high-order meta-path semantics and low-order relational interactions.

Despite the recent progress, learning discriminative HIN representations remains challenging because heterogeneous semantics manifest at multiple scales. Relation-based message passing preserves fine-grained typed-edge interactions but is inherently local; expanding the receptive field by stacking deeper layers may dilute discriminative signals and exacerbate the over-smoothing problem. In contrast, meta-path-based paradigms provide high-order semantics, yet they risk losing fine-grained local dependencies. In addition, prior methods generally underutilize topological structural cues in heterogeneous graphs for representation learning. Furthermore, as previously noted, integrating these paradigms is hindered by feature-space discrepancies, making naive fusion prone to semantic misalignment. Therefore, the central challenge is to effectively extract the fine-grained low- and high-order information and HIN topological semantics, and to progressively align these multi-scale features into a unified but discriminative consensus embedding for downstream tasks.

To address these limitations, we propose HINMOIT, a novel representation learning framework for heterogeneous information networks based on multi-order information extraction and interaction. For multi-order information extraction, we introduce dual-branch encoders to concurrently capture low-order structural relationships and high-order semantic correlations from complex heterogeneous graphs. The two encoders complement each other to enhance the mining of heterogeneity, effectively mitigating the over-smoothing issue inherent in low-order models and the fine-grained information loss prevalent in high-order models. For multi-order information interaction, we introduce a gated information interaction module to dynamically align and fuse multi-order features from the dual encoders. This mechanism effectively resolves semantic inconsistencies and facilitates the adaptive fusion of complementary information.

The contributions of this work are summarized as follows:

- We propose a novel representation learning framework for heterogeneous information networks. The framework integrates multi-order information extraction and interaction by leveraging dual encoders and a gated information interaction mechanism.
- For multi-order information extraction, the dual encoders are designed to learn both local and global heterogeneous graph information. The low-order relation-aware encoder captures fine-grained local structures, effectively mitigates the common over-smoothing problem associated with solely extracting low-order information. The high-order meta-graph encoder extracts broader semantic contexts through 1-hop meta-graph aggregation, thereby addressing the fine-grained semantic loss problem of only capturing high-order information.
- For multi-order information interaction, the gated information interaction module dynamically aligns features of the dual encoders within a unified space, resolving semantic inconsistencies and enabling complementary information fusion.
- Extensive validation on benchmark datasets demonstrates that HINMOIT outperforms state-of-the-art methods in node classification and clustering tasks.

The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 formalizes problem definitions, Section 4 details the architecture of HINMOIT, Section 5 presents experiments, and Section 6 concludes the paper.

2. Related works

2.1. Homogeneous graph representation learning

Homogeneous graph representation learning focuses on encoding nodes into low-dimensional embeddings while preserving structural and attribute information. Early graph representation methods employ matrix factorization and shallow embeddings. LINE [19] models first and second-order node proximities explicitly. Random-walk based approaches like DeepWalk [20] and node2vec [21] extend word2vec to graphs by generating node sequences via random walks and applying skip-gram for embedding learning.

The advent of Graph Neural Networks (GNNs) has shifted the paradigm toward end-to-end representation learning. Spatial-based models like GAT [22] leverage attention mechanisms for adaptive neighbor weighting, and GraphSAGE [23] introduces inductive learning through neighborhood sampling and aggregation. GIN [24] theoretically aligns message passing with the Weisfeiler-Lehman test for enhanced expressiveness. Spectral methods, such as GCN [25], implement spectral graph convolution through a localized first-order approximation. To address scalability, Cluster-GCN [26] partitions graphs into subgraphs for efficient batch training, and DropEdge [27] randomly removes edges during training to mitigate over-smoothing.

Recent advances focus on self-supervision and interpretability. GNNExplainer [28] identifies influential substructures by maximizing mutual information between predictions and subgraphs. PGExplainer [29] parameterizes subgraph importance to explain GNN predictions. Self-supervised methods, including DGI [30] and GraphCL [31], leverage contrastive learning to exploit intrinsic graph properties without labeled data; GMI [32] maximizes mutual information between node features and embeddings.

Beyond single type graphs, homogeneous embedding also arises in knowledge graphs and knowledge hypergraphs, where the objective shifts to modeling complex entity–relation interactions via tuple scoring. Representative works include HyCubE [33], which employs 3D circular convolution with a novel masking design to achieve efficient end-to-end n-ary tuple embedding. Similarly, HJE [34] integrates position-aware embeddings with joint 3D and relation-aware 2D convolutions to capture intricate tuple semantics. ConvD [35] proposes dynamic convolution kernels constructed from relations, weighted by attention mechanisms to enhance entity–relation interactions.

Although conventional GNNs have achieved remarkable success on homogeneous graphs, they implicitly assume a single node/edge type, aggregating neighboring information within a shared feature space. When directly applied to heterogeneous information networks (HINs) that contain multiple node types and relation types, such homogeneous aggregation schemes inevitably neglect type-specific semantics and fail to adequately model relation-dependent message passing. This limitation has motivated the development of heterogeneous graph representation learning frameworks.

2.2. Heterogeneous graph representation learning

Heterogeneous graph representation learning aims to preserve type-specific semantics and relation-dependent interactions in heterogeneous information networks, thereby overcoming the limitations of

homogeneous models on heterogeneous data. Accordingly, HGNNs can be broadly categorized into meta-path-based and relation-based methods.

For meta-path-based models, heterogeneous graphs are converted to homogeneous graphs via predefined semantic paths, so that homogeneous graph-related models can be used to process and aggregate the neighbor information of specific node types. They can naturally capture higher-order semantic information to avoid the over-smoothing problem caused by the convolution of multilayered graphs. Metapath2vec [36] adopts meta-path-guided random walks to capture structural and semantic relevance of different types of nodes and relations. HIN2Vec [37] extends skip-gram-based models by using random walks and negative sampling to jointly learn node embeddings and node-specific relationships. HAN [9] introduces hierarchical attention, including node-level attention to weigh neighboring nodes and semantic-level attention to weigh different meta-paths. MAGNN [38] further enhances the learning of different types of neighbor nodes within a meta-path by aggregating the entire meta-path instances into feature vectors. GTN [39] avoids predefined meta-paths by generating meaningful meta-paths with arbitrary lengths through soft selection of edges. CKD [40] performs collaborative knowledge extraction both from within and among meta-paths by modeling both local knowledge and global knowledge. Despite their effectiveness, most meta-path-based methods predominantly focus on meta-path-induced contexts while insufficiently modeling direct one-hop neighbors, which restricts fine-grained local semantic extraction and often results in coarse representations. Moreover, the aggregation process typically relies on feature-level attention, which is weakly topology-aware and fails to fully exploit deeper structural cues. The use of multiple meta-paths can also introduce semantic redundancy, and target-type-centric aggregation tends to under-represent the semantics of non-target node types.

Relation-based methods preserve heterogeneity through edge-type-aware aggregation. RGCN [41] aggregates the corresponding neighboring nodes through different types of relations, respectively. RSHN [42] integrates the original graph and the coarse line graph based on the relational structure perception, and embeds the nodes and edges into the heterogeneous graph without any prior knowledge such as meta-paths. HetGNN [43] uses restarted random walks to sample strongly correlated heterogeneous neighbors, encodes them with Bi-LSTM, and applies attention to weight different neighbor types. HGT [11] uses meta-relations to model heterogeneity with the help of a transformer to aggregate 1-hop neighbor information and automatically learns which meta-paths are more important. Simple-HGN [10] uses GAT [22] as its backbone to capture edge heterogeneity and addresses gradient vanishing and over-smoothing problems through learnable edge-type embeddings, residual connections and L2 normalization. HINormer [44] samples the context sequence of each node, adopts local structure and heterogeneous relation encoder, and enhances the node representation with the help of global attention mechanism of a transformer. SlotGAT [45] introduces semantic slots to address semantic confusion inherent in heterogeneous neighborhoods. Although relation-based models effectively handle typed-edge message passing, they primarily focus on local feature interactions and often fail to capture topology-informed higher-order dependencies without stacking multiple layers, which in turn exacerbates over-smoothing and optimization difficulties.

Recent hybrid or multi-scale HGNNs [16,17] attempt to integrate multi-order semantic information to mitigate the aforementioned issues. However, these approaches still exhibit the following critical limitations. First, they lack fine-grained mechanisms for extracting features across local-global scopes and remain insufficiently topology-aware. Moreover, hybrid models typically adopt a late fusion strategy, which overlooks the inherent discrepancies in feature scale and latent space distribution between different encoder outputs. Without explicit alignment and layer-wise interaction, such fusion tends to mix semantically non-commensurate representations, thereby undermining the complementarity of multi-view features. These limitations motivate our proposed layer-wise cross-view interaction scheme, designed to facilitate progressive alignment and dynamic information exchange between encoders.

3. Preliminaries

3.1. Heterogeneous information network

A heterogeneous information network (HIN) can be defined as $G = \{\mathcal{V}, \mathcal{E}, X, \phi, \psi\}$, where $\mathcal{V}$ represents the set of nodes, $\mathcal{E}$ represents the set of edges, $X \in \mathbb{R}^{|V| \times d_x}$ is the feature matrix, $\mathbf{x}_v \in \mathbb{R}^{d_x}$ is the feature vector of node $v \in \mathcal{V}$, and the attribute dimensionality d_x varies with the type of node v. Each node $v \in \mathcal{V}$ is associated with a node type $\phi(v) = t_i \in \mathcal{T}$, where $\mathcal{T}$ represents the node type set and ϕ is the node type mapping function. Each edge $e_{uv} \in \mathcal{E}$ is associated with an edge type $\psi(e_{uv}) = r_{ij} \in \mathcal{R}$, where $\mathcal{R}$ denotes the edge type set and ψ is the edge type mapping function. In the graph G, it holds that: $|\mathcal{T}| + |\mathcal{R}| > 2$. Let A_{t_i,t_j} denote the adjacency matrix between node types t_i and t_j, where $A_{t_i,t_j}[u][v] = 1$ indicates that there exists an edge between nodes u and v, such that $\phi(u) = t_i$ and $\phi(v) = t_j$.

3.2. Heterogeneous graph neural networks

Heterogeneous Graph Neural Networks (HGNNs) primarily utilize a layer-by-layer aggregation of neighboring information to derive target node representations. The node embeddings $\mathbf{h}_v^L \in \mathbb{R}^{d_L}$ are obtained from the graph structure and initial node features $\mathbf{h}_v^0 \in \mathbb{R}^{d_0}$ as:

$$\mathbf{h}_v^{(l)} = \mathrm{Aggr}^{(l)}\left(\left\{\mathbf{h}_u^{(l-1)} \mid u \in \mathcal{N}(v)\right\}; \theta_g^l\right) \tag{1}$$

where $\mathcal{N}(v)$ denotes the neighbors of node v, and $\mathrm{Aggr}(\cdot\,; \theta_g^l)$ represents the aggregation function at layer l, parameterized by θ_g^l. This function can be implemented using attention-based aggregation, average pooling, or other methods.

As a specialized class of GNNs, HGNNs are designed to model complex heterogeneous graphs. These methods often involve multilayered relational processing such as the representation in RGCN:

$$\mathbf{h}_v^{(l)} = \sigma\left(\sum_{r \in \mathcal{R}} \sum_{u \in \mathcal{N}_v^r} \frac{1}{c_{v,r}} \mathbf{W}_r^{(l-1)} \mathbf{h}_u^{(l-1)} + \mathbf{W}_0^{(l-1)} \mathbf{h}_v^{(l-1)}\right) \tag{2}$$

where $\mathcal{N}_v^r$ is the set of neighbors of node v connected via relation r, and the aggregation of neighboring nodes is differentiated by edge types. $c_{v,r}$ denotes the number of neighbors of node v under relation r, and $\mathbf{W}_r$ and $\mathbf{W}_0$ represent the weight matrices for relation r and self-loop, respectively.

The primary goal of HGNNs is to map nodes in a HIN to low-dimensional feature vectors while preserving graph-specific properties. Formally, this is achieved by defining a mapping function $\mathcal{F} : \mathbf{x}_v \to \mathbb{R}^{d_x}$, where $d_x \ll |\mathcal{V}|$.

4. Proposed method

4.1. Overview of HINMOIT

As illustrated in Fig. 1, the proposed framework HINMOIT comprises three core modules, including the relation-aware encoder, the meta-graph encoder and the multi-order information interaction module. The relation-aware encoder captures low-order local structural patterns and feature information proximal to the target node, while the meta-graph encoder extracts high-order global semantics and topological dependencies. These dual encoders operate in parallel to aggregate neighborhood information from complementary perspectives. Both encoders adopt an attention mechanism to weight neighbor contributions during aggregation and leverage structural semantic properties to guide topology-aware neighbor selection using aggregation coefficients. Specifically, the relation-aware encoder defines structural semantics as node degree,

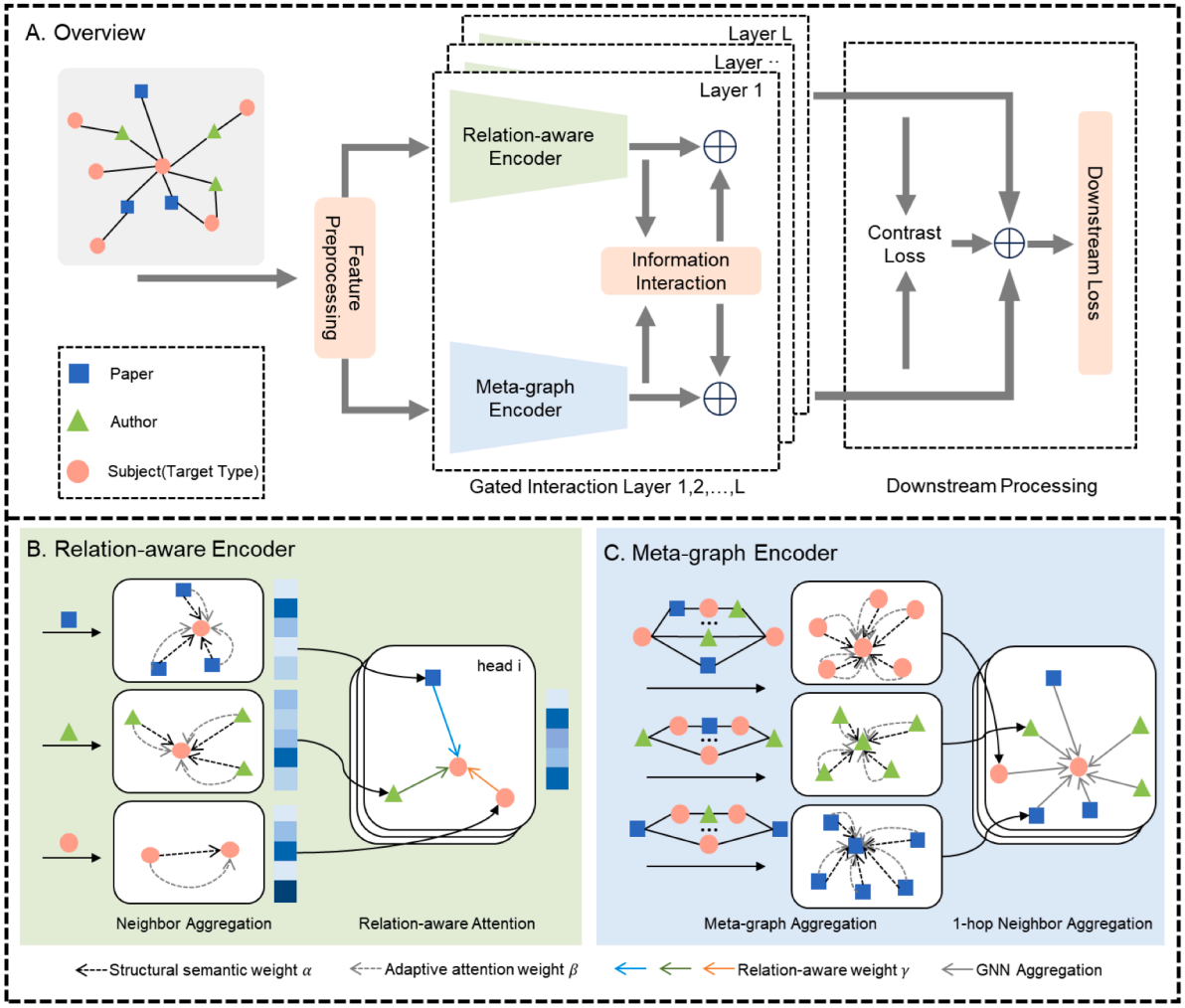


Fig. 1. Illustration of the proposed HINMOIT. (A) Overview of HINMOIT. The HINMOIT framework comprises three key modules, including relation-aware encoder, meta-graph encoder and multi-order information interaction module. (B) relation-aware encoder. (C) meta-graph encoder.

whereas the meta-graph encoder utilizes the number of edges between nodes as its structural semantic measure. Subsequently, to resolve the inherent semantic misalignment between representations captured at different orders, the multi-order information interaction module employs a gated mechanism to progressively align the feature spaces. This iterative refinement promotes the learning of consensus representations through constructive feature interaction, thereby significantly enhancing fusion efficacy. Finally, to enforce consistency across views, a contrastive learning objective is applied to the fused features in the final layer, which is jointly optimized with the downstream supervised task loss.

Distinct from mainstream meta-path-based approaches such as HAN [9] and MAGNN [38], which typically encapsulate heterogeneous semantics within a single encoder via meta-path-induced views, HINMOIT adopts an explicit order-specialized dual-encoder architecture. In comparison to the dual-channel framework of HeCo [46], our relation-aware encoder offers a more fine-grained decoupling of heterogeneity. It decomposes the HIN into relation-specific bipartite subgraphs, where neighbor importance is determined dynamically by both structural semantic weights and adaptive attention scores. A residual connection is introduced to directly preserve the target node's own features, effectively mitigating information loss during local aggregation. Simultaneously, the meta-graph encoder captures high-order semantic dependencies by synthesizing multiple meta-path-induced adjacency matrices, without being restricted to only the target node type. It similarly employs the dual-weight scheme to evaluate node importance, and then performs only 1-hop neighborhood aggregation on the original HIN to obtain high-order semantic representations, while also retaining node-wise residual connections to preserve the original features. By leveraging the extracted local and global contexts, a gating mechanism is employed to progressively align the feature spaces across views and learn consensus information, further strengthening the semantic representations from different perspectives.

4.2. Relation-aware encoder

This module captures low-order structural information surrounding target nodes. While traditional GNNs leverage multi-layer convolutions for neighbor information aggregation, they often neglect structural semantic information essential for modeling heterogeneity, limiting neighbor selection reliability. To address this limitation, our encoder enables relation-specific fine-grained aggregation through three steps. First, according to the relation types, the heterogeneous information network is decomposed into different bipartite subgraphs. Then, it incorporates an attention mechanism weighted by node content features and structural semantic guidance using node degrees to quantify topological influence. Finally, relation-aware aggregation is performed over the multiple relations associated with each node.

Given different node types reside in disparate feature spaces with potentially different dimensionalities, direct neighbor aggregation might be infeasible. We therefore project all node features into a unified d-

dimensional space through type-specific linear transformations:

$$\mathbf{h}_v = \sigma\left(\mathbf{W}_{\phi(v)}\mathbf{x}_v + \mathbf{b}_{\phi(v)}\right) \tag{3}$$

where σ is the GELU activation function, $\mathbf{W}_{\phi(v)} \in \mathbb{R}^{d\times d_x}$ and $\mathbf{b}_{\phi(v)} \in \mathbb{R}^d$ respectively are the learnable parameter matrix and bias term, and $\mathbf{x}_v \in \mathbb{R}^{d_x}$ is the raw feature vector of node v.

1) Relation-aware Transformation: To better capture structural and relational heterogeneity, we incorporate attention mechanisms and design fine-grained aggregation strategies tailored to each relation type. According to the relation types, the heterogeneous information network is decomposed into different bipartite subgraphs. As different relations around the target node represent specific semantic knowledge, we apply a relation-aware transformation to encode edge semantics:

$$\tilde{\mathbf{h}}_v^{(l,k)} = \mathbf{W}_{\phi(v)}^{(l,k)} \cdot \mathbf{h}_v^{(l)} \tag{4}$$

$$\tilde{\mathbf{h}}_u^{(l,k)} = \mathbf{W}_{\psi(e_{u,v})}^{(l,k)} \cdot \mathbf{h}_u^{(l)} \tag{5}$$

where $\mathbf{h}_v^{(l)}$ and $\tilde{\mathbf{h}}_v^{(l,k)}$ are the feature vectors of the target node v before and after the transformation, $\mathbf{h}_u^{(l)}$ and $\tilde{\mathbf{h}}_u^{(l,k)}$ denote the feature vectors of the neighbor node u, before and after the transformation, respectively, with respect to the target node v under the edge relation $\psi(e_{u,v})$, and $\mathbf{W}_{\psi(e_{u,v})}^{(l,k)}$ is the trainable parameter matrix for relation $\psi(e_{u,v})$ at layer l and attention head k.

2) Dual Neighbor Weighting: For each bipartite subgraph, we compute the influence of neighbor u on target node v from two complementary perspectives:

The content-based attention coefficient $\beta_{(u,v)}^{(l,k)}$ that accounts for node features is computed as:

$$\beta_{(u,v)}^{(l,k)} = \frac{\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}^{(l,k)^{\mathbf{T}}}\left[\tilde{\mathbf{h}}_u^{(l,k)} \,\|\, \tilde{\mathbf{h}}_v^{(l,k)}\right]\right)\right)}{\sum_{j\in\mathcal{N}_v^r}\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}^{(l,k)^{\mathbf{T}}}\left[\tilde{\mathbf{h}}_j^{(l,k)} \,\|\, \tilde{\mathbf{h}}_v^{(l,k)}\right]\right)\right)} \tag{6}$$

where $\beta_{(u,v)}^{(l,k)}$ is the attention coefficient of the k-th attention head of the l-th layer between the target node v and the neighbor node u, $\mathcal{N}_v^r$ represents the set of neighbor nodes of the target node v through the relation r, and $\mathbf{a}^{(l,k)}$ is the parameterized attention vector.

From the topological perspective, the importance of different neighbor nodes is computed based on the indegree of the target node v and the outdegree of the neighboring node u:

$$\alpha_{(u,v)}^{(l,k)} = \frac{1}{\sqrt{|\mathcal{N}_u| \cdot |\mathcal{N}_v|}} \tag{7}$$

where $|\mathcal{N}_v|$ is the number of neighbors of node v. By introducing the degree into the node aggregation, more structural semantics and heterogeneity information of the target node is captured.

The feature vector $\mathbf{h}_{(v,r)}^{(l,k)}$ of the target node v in the bipartite subgraph based on the relation r is obtained by weighted summation:

$$\mathbf{h}_{(v,r)}^{(l,k)} = \sum_{u\in\mathcal{N}_v^r}\left(\lambda_{ra}\cdot\alpha_{(u,v)}^{(l,k)} + (1-\lambda_{ra})\cdot\beta_{(u,v)}^{(l,k)}\right)\cdot\tilde{\mathbf{h}}_u^{(l,k)} \tag{8}$$

where λ_{ra} is a hyperparameter to regulate the weight of the heterogeneous proximity.

3) Relation-aware Aggregation: To aggregate the neighborhood information across distinct relation types while preserving local structural patterns, we compute relation importance weights through an attention mechanism:

$$\gamma_{(v,r)}^{(l,k)} = \frac{\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}_s^{(l,k)^{\mathbf{T}}}\left[\mathrm{Norm}\left(\tilde{\mathbf{h}}_v^{(l,k)}\right) \,\|\, \mathrm{Norm}\left(\mathbf{h}_{(v,r)}^{(l,k)}\right)\right]\right)\right)}{\sum_{j\in\mathcal{R}_v}\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}_s^{(l,k)^{\mathbf{T}}}\left[\mathrm{Norm}\left(\tilde{\mathbf{h}}_v^{(l,k)}\right) \,\|\, \mathrm{Norm}\left(\mathbf{h}_{(v,j)}^{(l,k)}\right)\right]\right)\right)} \tag{9}$$

where $\gamma_{(v,r)}^{(l,k)}$ is the relation-aware attention coefficient for relation r at layer l, head k. $\mathbf{a}_s^{(l,k)}$ is the learnable attention vector, $\mathrm{Norm}(\cdot)$ denotes L2 normalization, and $\mathcal{R}_v$ is the set of edge relation types of the node v.

Based on the importance of different relations, relation-specific embeddings are combined across attention heads. Meanwhile, residual connections are introduced between layers to retain the self-information of nodes, and L2 normalization is used after each layer encoder to rescale the features and maintain the consistency of the inter-layer scales:

$$\tilde{\mathbf{h}}_v^{(l+1)} = \Big\|_{k=1}^{K}\left(\sum_{r\in\mathcal{R}_v}\gamma_{(v,r)}^{(l,k)}\cdot\mathbf{h}_{(v,r)}^{(l,k)}\right) \tag{10}$$

$$\mathbf{h}_v^{(l+1)} = \mathrm{Norm}\left(\sigma\left(\tilde{\mathbf{h}}_v^{(l+1)} + \mathbf{h}_v^{(l)}\right)\right) \tag{11}$$

where $\|_{k=1}^{K}$ denotes the multiple concatenation operation and σ is the GELU activation function. The layer output is denoted as $\mathbf{h}_{ra}^{(l+1)} = \mathbf{h}_v^{(l+1)}$ for subsequent processing, yielding the representation sequence $\{\mathbf{h}_{ra}^{(1)}, \mathbf{h}_{ra}^{(2)}, \ldots, \mathbf{h}_{ra}^{(n)}\}$.

4.3. Meta-graph encoder

In heterogeneous graph networks, distant nodes also exert influence on target node representations. While conventional approaches capture long-range dependencies through multi-layer graph convolutions, they frequently suffer from over-smoothing and over-compression problems. Although meta-path-based methods model high-order relationships, they predominantly focus on target node types, ignoring the remaining types of neighboring nodes. To address these limitations, our module constructs type-specific meta-graphs by fusing all meta-paths associated with each neighbor node type. After performing heterogeneous aggregation on each meta-graph, target node embeddings are derived through efficient 1-hop neighbor aggregation. This approach simultaneously mitigates the over-smoothing problem while preserving heterogeneous information across node types through compressed high-order dependencies.

1) Dual-weighted Meta-graph Aggregation: Meta-paths play a key role in maintaining long-range dependencies, and how to further enhance meta-paths to capture richer semantic information in heterogeneous graphs remains an open question. We augment meta-paths by fusing multiple meta-paths to construct a meta-graph. For a node type $\phi(v_i) = t_i \in \mathcal{T}$, all its meta-paths are denoted as $\mathcal{P}^{(t_i)} = \{p_1^{(t_i)}, p_2^{(t_i)}, \ldots, p_n^{(t_i)}\}$, where each meta-path $p_s^{(t_i)}$ corresponds to an adjacency matrix $A_s^{(t_i)}$, and the set of all adjacency matrices is denoted as $\mathcal{A}^{(t_i)} = \{A_1^{(t_i)}, A_2^{(t_i)}, \ldots, A_n^{(t_i)}\}$. The meta-graph structure is defined as:

$$\mathbf{G}^{(t_i)} = \sum_{A_s^{(t_i)}\in\mathcal{A}^{(t_i)}} A_s^{(t_i)} \tag{12}$$

where $\mathbf{G}^{(t_i)}$ is the meta-graph adjacency matrix corresponding to node type t_i.

As in relation-aware encoders, aggregation in meta-graphs is based on a dual weighting mechanism that integrates both semantic and structural information:

$$\beta_{(u,v)}^{(l,t_i)} = \frac{\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}_{t_i}^{l^{\mathbf{T}}}\left[\mathbf{h}_u^l \,\|\, \mathbf{h}_v^l\right]\right)\right)}{\sum_{j\in\mathcal{N}_v^{(t_i)}}\exp\left(\mathrm{LeakyReLU}\left(\mathbf{a}_{t_i}^{l^{\mathbf{T}}}\left[\mathbf{h}_j^l \,\|\, \mathbf{h}_v^l\right]\right)\right)} \tag{13}$$

$$\alpha_{(u,v)}^{(l,t_i)} = \frac{\exp\left(\mathbf{G}^{(t_i)}[u][v]\right)}{\sum_{(i,j)\in\mathbf{G}^{(t_i)}}\exp\left(\mathbf{G}^{(t_i)}[i][j]\right)} \tag{14}$$

$$\mathbf{h}_v^{(l,t_i)} = \sum_{u\in\mathcal{N}_v^{(t_i)}}\left(\lambda_{\mathrm{mg}}\cdot\alpha_{(u,v)}^{(l,t_i)} + (1-\lambda_{\mathrm{mg}})\cdot\beta_{(u,v)}^{(l,t_i)}\right)\cdot\mathbf{h}_u^l \tag{15}$$

where $\beta_{(u,v)}^{(l,t_i)}$ is the attention coefficient at layer l between the target node v and neighbor node u, $\mathcal{N}_v^{(t_i)}$ is the neighboring set of v in the meta-graph $\mathbf{G}^{(t_i)}$, $\mathbf{a}_{t_i}^l$ is the parameterized attention vector, $\alpha_{(u,v)}^{(l,t_i)}$ is the topological

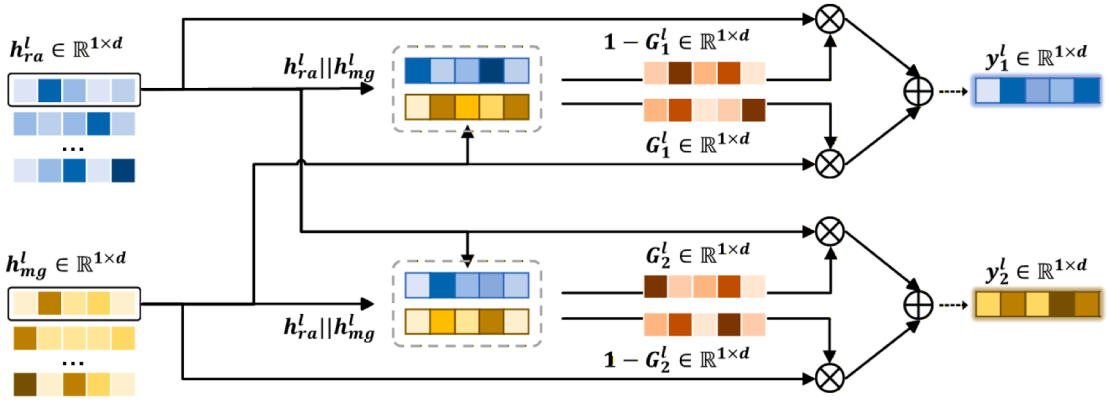


Fig. 2. The dynamic gated interaction mechanism. h_{ra}^{l} and h_{mg}^{l} represent the layer-wise outputs of the dual encoders, while y_1^l and y_2^l denote the representations obtained after fusion and interaction.

proximity between the target node v and the neighbor node u, $\mathbf{G}^{(t_i)}[u][v]$ denotes the edge count between u and v in $\mathbf{G}^{(t_i)}$, λ_{mg} is a balance hyperparameter, and $\mathbf{h}_v^{(l,t_i)}$ is the final meta-graph embedding, which is labeled as $\mathbf{h}_v^{(\text{mg})}$ for subsequent analysis.

2) 1-hop Neighbor Aggregation: Based on meta-graph aggregation, high-order information from all types of neighboring nodes can be captured. Therefore, the target node only needs to aggregate the high-order information of its 1-hop neighbors:

$$\tilde{\mathbf{h}}_v^{(l+1)} = \text{Aggr}^{(l)}\left(\left\{\mathbf{h}_u^{(\text{mg})} : u \in \mathcal{N}(v)\right\}; \theta_g^l\right) \tag{16}$$

where $\mathcal{N}(v)$ is the neighboring set of node v in the original heterogeneous graph, and θ_g^l is the parameter at layer l.

To preserve the original features, residual connections are employed across layers, and each encoder layer is followed by L2 normalization for inter-layer scale consistency:

$$\mathbf{h}_v^{(l+1)} = \text{Norm}\left(\sigma\left(\tilde{\mathbf{h}}_v^{(l+1)} + \mathbf{h}_v^{(l)}\right)\right) \tag{17}$$

The integration of a multi-head mechanism with the above operation helps to stabilize training and improve robustness. The output $\mathbf{h}_v^{(l+1)}$ of each layer of the meta-graph encoder is denoted as $\mathbf{h}_{\text{mg}}^{(l+1)}$, i.e., $\{\mathbf{h}_{\text{mg}}^{(1)}, \mathbf{h}_{\text{mg}}^{(2)}, \ldots, \mathbf{h}_{\text{mg}}^{(n)}\}$.

4.4. Multi-order information interaction

The relation-aware and meta-graph encoders capture complementary low-order structural patterns and high-order dependencies, respectively. However, their distinct inductive biases yield different latent spaces, and direct fusion may yield suboptimal representations due to unaligned spaces and semantic incompatibility. As shown in Fig. 2, based on a gated interaction mechanism, this module enables gradual feature space alignment and semantic feature interaction, to enhance the feature fusion process and learn consensus information:

$$G_1 = \sigma\left(w_1\left(\mathbf{H}_{ra} \,\|\, \mathbf{H}_{mg}\right)\right) \tag{18}$$

$$G_2 = \sigma\left(w_2\left(\mathbf{H}_{ra} \,\|\, \mathbf{H}_{mg}\right)\right) \tag{19}$$

$$y_1 = G_1 \cdot \mathbf{H}_{mg} + (1 - G_1) \cdot \mathbf{H}_{ra} \tag{20}$$

$$y_2 = G_2 \cdot \mathbf{H}_{ra} + (1 - G_2) \cdot \mathbf{H}_{mg} \tag{21}$$

where $\mathbf{H}_{ra}$ denotes $\{\mathbf{h}_{ra}^{(1)}, \mathbf{h}_{ra}^{(2)}, \ldots, \mathbf{h}_{ra}^{(n)}\}$, $\mathbf{H}_{mg}$ denotes $\{\mathbf{h}_{mg}^{(1)}, \mathbf{h}_{mg}^{(2)}, \ldots, \mathbf{h}_{mg}^{(n)}\}$, w is the trainable parameter, σ is the activation function, and y_1 and y_2 are the fused encoder-specific consensus representations.

To preserve encoder-specific features while incorporating cross-order consensus, we apply residual fusion:

$$\tilde{\mathbf{h}}_{ra}^{(l)} = \lambda_{G1} \cdot y_1 + (1 - \lambda_{G1}) \cdot \mathbf{h}_{ra}^{(l)} \tag{22}$$

$$\tilde{\mathbf{h}}_{mg}^{(l)} = \lambda_{G2} \cdot y_2 + (1 - \lambda_{G2}) \cdot \mathbf{h}_{mg}^{(l)} \tag{23}$$

where λ_{G1} and λ_{G2} are weight hyperparameters of different semantic information, and the fused outputs $\tilde{\mathbf{h}}_{ra}^{(l)}$ and $\tilde{\mathbf{h}}_{mg}^{(l)}$ serve as inputs to subsequent encoder layers.

Here, we provide a theoretical analysis for how the proposed interaction mechanism alleviates representation collapse.

Let $\tilde{\mathbf{H}} \in \mathbb{R}^{|\mathcal{V}|\times d}$ be the final node embedding matrix (row-wise L_2 normalized). Representation collapse occurs when the embeddings become indistinguishable across nodes, resulting in a Gram matrix $\mathbf{K} = \tilde{\mathbf{H}}\tilde{\mathbf{H}}^{\top}$ that approaches an all-ones matrix. Spectrally, this implies that the embedding covariance becomes highly anisotropic, with an effective rank close to 1.

Our interaction is a *residual correction* rather than a hard replacement. From Eq. (20)-(23), the updated representation for the relation-aware branch can be rewritten as

$$\tilde{\mathbf{H}}_{ra}^{(l)} = \mathbf{H}_{ra}^{(l)} + \lambda_{G1}\, \mathbf{G}_1^{(l)} \odot \left(\mathbf{H}_{mg}^{(l)} - \mathbf{H}_{ra}^{(l)}\right), \tag{24}$$

and similarly for $\tilde{\mathbf{H}}_{mg}^{(l)}$. Eq. (24) shows that (i) each encoder keeps an identity path $\mathbf{H}_{ra}^{(l)}$ (resp. $\mathbf{H}_{mg}^{(l)}$), and (ii) cross-order information enters as a bounded correction controlled by λ_{G1} and the data-dependent gate $\mathbf{G}_1^{(l)}$ (Eq. (18)). Therefore, to collapse the representation at layer l, the model would need both encoders to collapse and gates to saturate in a way that fully overwrites the identity path. In contrast, direct concatenation/averaging can erase encoder-specific variance in one step. Empirically, the gate learns to down-weight the branch that becomes less informative (e.g., over-smoothed), which further reduces the chance that both branches collapse simultaneously.

To maximize consistency between cross-order representations, contrast learning is further applied to the last layer, using InfoNCE as the cross-view contrast loss:

$$\mathcal{L}_{cl} = -\sum_{i\in\mathcal{V}} \log \frac{\exp\left(\text{sim}\left(\tilde{\mathbf{h}}_{ra_i}^{(L)}, \tilde{\mathbf{h}}_{mg_i}^{(L)}\right)/\tau\right)}{\sum_{j\in\mathcal{V}} \exp\left(\text{sim}\left(\tilde{\mathbf{h}}_{ra_i}^{(L)}, \tilde{\mathbf{h}}_{mg_j}^{(L)}\right)/\tau\right)} \tag{25}$$

where $\tilde{\mathbf{h}}_{ra_i}^{(L)}$ and $\tilde{\mathbf{h}}_{mg_i}^{(L)}$ are the embeddings of node i at the last layer of the two encoders, $\text{sim}(\cdot,\cdot)$ is the cosine similarity function, and τ is a temperature hyperparameter.

We further analyze how L_{cl} regularizes the embedding spectrum to prevent collapse. The InfoNCE objective (Eq. (25)) can be written using the cross-view similarity matrix

$$\mathbf{S} = \frac{1}{\tau}\,\text{sim}\left(\tilde{\mathbf{H}}_{ra}^{(L)}, \tilde{\mathbf{H}}_{mg}^{(L)}\right) \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|} \tag{26}$$

where the (i,j)-th entry is the cosine similarity between node i in view ra and node j in view mg. Eq. (25) is exactly the cross-entropy loss that classifies the positive pair (i,i) against negative pairs (i,j) $(j \neq i)$, thus it

Table 1
Statistics of the datasets.

Datasets	Nodes	Edges	Node Types	Edge Types	Target	Classes	Target Attributes
DBLP	26,128	239,566	4	6	author	4	334
IMDB	21,420	86,642	4	6	movie	5	3489
ACM	10,942	547,872	4	8	paper	3	1902

explicitly encourages $\mathbf{S}$ to be diagonally dominant: increasing S_{ii} while suppressing off-diagonal S_{ij}. This has a spectral implication: making off-diagonal similarities small forces embeddings of different nodes to spread out, which increases the effective rank and alleviates anisotropy. In the extreme collapsed case where all embeddings are identical, $\mathbf{S}$ becomes approximately constant and cannot be diagonally dominant, yielding a large loss and non-zero gradients that push the model away from the rank-one solution.

4.5. Training objective

For the node-level representation learning, we adopt the node classification task as the downstream task. The final node representations integrate both encoders' outputs through weighted fusion. Next, the class distribution is predicted using a linear layer function parameterized by θ_{pre}. The end-to-end framework is optimized through the total loss $\mathcal{L}$:

$$\mathbf{H} = \lambda_1 \cdot \tilde{\mathbf{h}}_{ra}^{(L)} + (1-\lambda_1) \cdot \tilde{\mathbf{h}}_{mg}^{(L)} \tag{27}$$

$$\hat{y}_i = \text{Linear}(\mathbf{H}_i; \theta_{\text{pre}}) \tag{28}$$

$$\mathcal{L}_{\text{nc}} = \sum_{i \in \mathcal{V}_{\text{train}}} \text{CE}(y_i, \hat{y}_i) \tag{29}$$

$$\mathcal{L} = \mathcal{L}_{\text{nc}} + \lambda_2 \cdot \mathcal{L}_{\text{cl}} \tag{30}$$

where λ_1 and λ_2 are the hyperparameter weights, $\hat{y}_i \in \mathbb{R}^C$ is the predicted label, and C is the number of categories.

5. Results

5.1. Experimental setups

Datasets. We validate our model HINMOIT on three publicly available benchmark datasets, including two academic citation datasets (DBLP and ACM) and a movie dataset (IMDB). Table 1 summarizes the statistics of the datasets. For the datasets that do not contain node attributes, we initialize node features with one-hot vectors of dimension $|\mathcal{V}|$.

Baselines. To comprehensively evaluate HINMOIT against state-of-the-art methods, we compare with 15 baseline models for node classification. These baselines include:

- Homogeneous GNNs: GCN [25] and GAT [22].
- Meta-path-based HGNNs: RGCN [41], HetGNN [43], HAN [9], GTN [39] and MAGNN [38].
- Meta-path-free HGNNs: RSHN [42], HetSANN [47], HGT [11], Simple-HGN [10], HINormer [44], SlotGAT [45], Tagformer [48] and AGHINT [49].

Hyperparameters for all baselines follow their original publications to ensure a fair comparison.

Settings and Parameters. We conduct multi-class node classification on DBLP and ACM, and multi-label classification on IMDB. Following the processing pipeline in HGB [10], the nodes are randomly divided into 24:6:70 as the training, validation and test sets. Performance is evaluated using Micro-F1 and Macro-F1, with mean ± std reported from 5 independent experiments. We use Adam as the optimizer, LeakyReLU with a negative slope of 0.2 as the activation function, and an early stopping mechanism with a patience value of 40 to prevent model overfitting. The learning rate is set to 1e-4, the dropout rate and the number of attention heads are set to 0.2 and 4 in the relation-aware encoder, and 0.1 and 2 in the meta-graph encoder, respectively. The gated interaction layer is set to 2, where the number of layers in each cascade block for the relation-aware encoder is 2, the number of inter-class layers for the meta-graph encoder is set to [2,3], the number of intra-class layers is set to [1,2]. The hidden layer dimension is set to 512 for the ACM dataset and 256 for the other datasets. Details of the meta-graph configuration are provided in Table 2.

5.2. Performance comparison

To evaluate the node representation capability of the proposed method HINMOIT, we conduct two downstream tasks across three benchmark datasets. Specifically, we conduct multi-class node classification on DBLP and ACM, and multi-label classification on IMDB. Since clustering is only performed for multi-class classification in this experiment, the node clustering task is conducted on the DBLP and ACM datasets. These tasks comprehensively assess discriminative feature learning with different evaluation metrics.

As shown in Table 3, HINMOIT achieves state-of-the-art performance in the node classification task, surpassing all baselines in both Micro-F1 and Macro-F1 metrics. In detail, HINMOIT significantly outperforms both meta-path-based and meta-path-free HGNNs. The results validate the effectiveness of the multi-order information extraction strategy, which captures heterogeneous information from both global and local views. Compared to the graph transformer-based model, HINMOIT achieves significantly better performance than HINormer on both ACM and IMDB. Specifically, HINMOIT gains 4.5% Macro-F1 improvement on IMDB, demonstrating superior heterogeneity modeling compared to attention-only models. HINMOIT also outperforms the SlotGAT model in both metrics, confirming the gated interaction mechanism enables semantic complementarity and prevents semantic confusion through progressive feature space alignment. Compared to meta-path-based methods [9,38,39], HINMOIT achieves a significant performance advantage due to its ability to capture finer-grained semantic information, with particularly notable improvements on the IMDB dataset. Compared to Simple-HGN, HINMOIT exhibits a 1.6% and 6.3% improvement in the Macro-F1 metrics on DBLP and IMDB, respectively. Unlike Simple-HGN, which adds heterogeneous information through edge-type embedding, HINMOIT introduces a dual-encoder architecture to extract node heterogeneity from a more unified perspective, leading to more discriminative node representations.

For further validation, we conduct node clustering on DBLP and ACM. Here, cluster assignments are derived from K-means applied to optimal validation embeddings. The clustering results are evaluated by two widely used metrics, NMI and ARI. As shown in Table 4, HINMOIT significantly outperforms the baseline models in terms of NMI and ARI. Specifically, compared to HINormer, HINMOIT achieves improvements of 5.1% and 3.5% in NMI and ARI, respectively. When compared to the meta-path-based model HAN, it exhibits even larger improvement of 20.8% and 22.5% on the same metrics. These results highlight HINMOIT's superior ability to capture complex heterogeneity and effectively distinguish features across different clusters. These substantial gains confirm our model effectively captures complex heterogeneous structures and diverse edge relations, generating highly discriminative

Table 2
Statistics of the meta-graphs.

Dataset	Meta-graph
DBLP	[A-P-A, A-P-T-P-A, A-P-V-P-A], [P-A-P, P-T-P]
IMDB	[M-D-M, M-A-M, M-K-M], [D-M-D, D-M-A-M-D], [A-M-A, A-M-D-M-A], [K-M-K, K-M-D-M-K]
ACM	[P-P-P, P-A-P, P-C-P, P-K-P], [A-P-P-A, A-P-C-P-A], [C-P-C, C-P-P-C, C-P-A-P-C], [K-P-K, K-P-A-P-K]

Table 3
Performance comparison of different methods on node classification in terms of Micro F1 and Macro F1. The best results are denoted in bold, with the second-best results underlined. The error bars (±) indicate the standard deviation of the results over five independent trials.

Dataset	DBLP		ACM		IMDB	
Model \ Metrics	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
GCN	91.47±0.34	90.84±0.32	92.12±0.23	92.17±0.24	64.82±0.64	57.88±1.18
GAT	93.39±0.30	93.83±0.27	92.19±0.93	92.26±0.94	64.86±0.43	58.94±1.35
RGCN	92.07±0.50	91.52±0.50	91.41±0.75	91.55±0.74	62.05±0.15	58.85±0.26
HetGNN	92.33±0.41	91.76±0.43	86.05±0.25	85.91±0.25	51.16±0.65	48.25±0.67
HAN	92.05±0.62	91.67±0.49	90.79±0.43	90.89±0.43	64.63±0.58	57.74±0.96
GTN	93.97±0.54	93.52±0.55	91.20±0.71	91.31±0.70	65.14±0.45	60.47±0.98
MAGNN	93.76±0.45	93.28±0.51	90.77±0.65	90.88±0.64	64.67±1.67	56.49±3.20
RSHN	93.81±0.55	93.34±0.58	90.32±1.54	90.50±1.51	64.22±1.03	59.85±3.21
HetSANN	80.56±1.50	78.55±2.42	89.91±0.37	90.02±0.35	57.68±0.44	49.47±1.21
HGT	93.49±0.25	93.01±0.23	91.00±0.76	91.12±0.76	67.20±0.57	63.00±1.19
Simple-HGN	94.46±0.22	94.01±0.24	93.35±0.45	93.42±0.44	67.36±0.57	63.53±1.36
HINormer	94.94±0.21	94.57±0.23	92.12±0.27	92.19±0.27	67.83±0.34	64.65±0.53
SlotGAT	95.31±0.19	94.95±0.20	94.06±0.22	93.99±0.23	68.54±0.33	64.05±0.60
Tagformer	95.35±0.16	94.99±0.19	94.08±0.25	94.15±0.25	68.69±0.57	64.68±0.66
AGHINT	95.47±0.13	95.12±0.14	93.98±0.34	94.04±0.35	69.30±0.23	66.49±0.30
HINMOIT	**95.85±0.27**	**95.49±0.22**	**94.12±0.43**	**94.19±0.23**	**69.75±0.47**	**67.56±0.34**

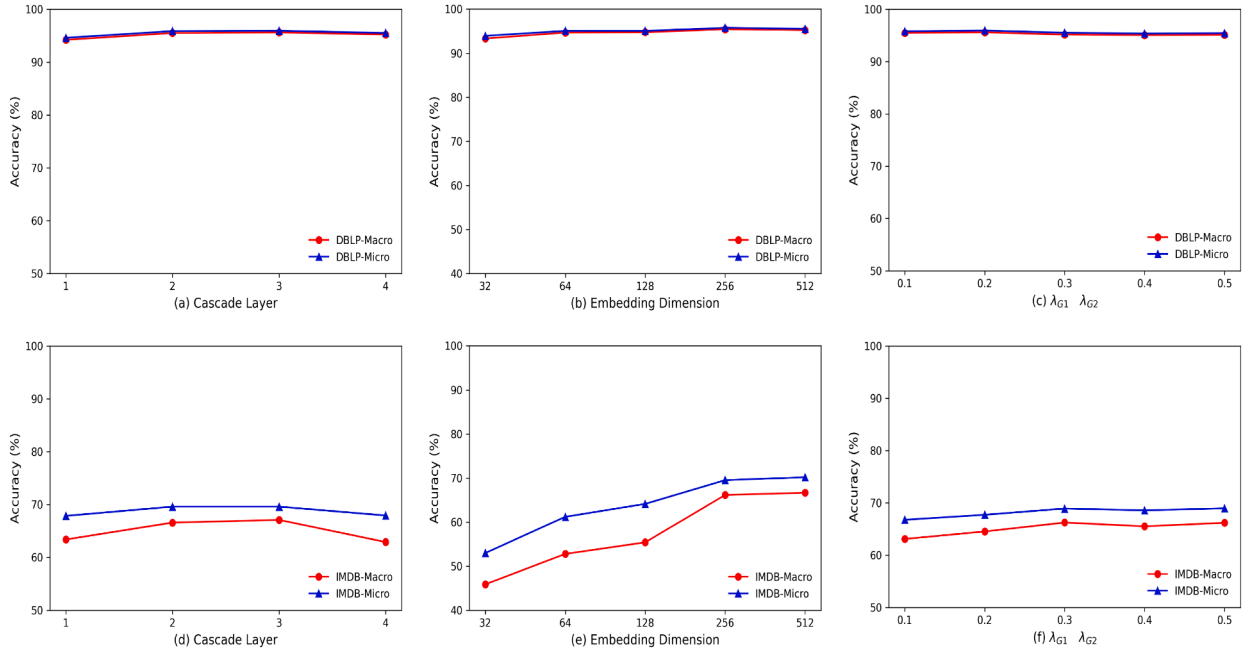


Fig. 3. Hyperparameters analysis of cascade layer, embedding dimension, $\lambda_G 1$ and $\lambda_G 2$ on DBLP and IMDB datasets. Results for DBLP are in the top row ((a)–(c)), and results for IMDB are in the bottom row ((d)–(f)).

representations that preserve both structural and semantic relationships within the graph.

5.3. Parameters sensitivity

We evaluate the sensitivity of key hyperparameters in HINMOIT. The results are presented in Fig. 3. Specifically, the Cascade Layer refers to the number of cascade blocks in the dual-encoder architecture, followed by an information interaction unit. The number of cascade blocks in our dual-encoder architecture demonstrates optimal performance at depths of 2–3 layers. The accuracy of HINMOIT shows a strong positive correlation with the dimensionality of the hidden layers across all encoders. Specifically, increasing dimensions from 32 to 256 improves the accuracy of HINMOIT by about 14–20% on the IMDB dataset, demonstrating that expanded representations better capture nuanced heterogeneous patterns and complex structural semantics. λ_{G1} and λ_{G2} serve as trade-off coefficients that regulate the integration of consensus information between the original encoder outputs and the fused representations in

Table 4
Performance comparison of different methods on node clustering in terms of NMI and ARI. The best results are denoted in bold, with the second-best results underlined.

Dataset	DBLP		ACM	
Model \ Metrics	NMI	ARI	NMI	ARI
GCN	66.72	73.65	69.27	74.59
GAT	78.95	83.96	69.18	73.72
HAN	78.70	84.40	64.36	68.15
MAGNN	80.24	85.72	71.75	76.37
HGT	77.85	84.01	64.48	67.64
Simple-HGN	79.27	83.54	<u>76.72</u>	<u>81.22</u>
HINormer	<u>81.11</u>	<u>86.64</u>	74.65	79.14
HINMOIT	**85.25**	**89.68**	**77.74**	**83.49**

the gated interaction module. While their influence is limited on the DBLP dataset, higher coefficient values yield improved performance on the IMDB dataset.

5.4. Ablation study

To investigate the effectiveness of each component in HINMOIT, we respectively conduct the ablation study for node classification task and node clustering task. Specifically, we investigate the following four variants:

- w/o RA: The relation-aware encoder is removed and replaced by a plain GCN.
- w/o MG: The meta-graph encoder is substituted with a plain GCN.
- w/o Contrast: Contrastive learning is not applied in the last layer of the dual-channel encoder.
- w/o Interaction: The gated interaction module is removed.

The evaluated performance of the node classification task is represented in Table 5. From the table, we observe that the HINMOIT model outperforms all ablation variants, affirming the essential role of each component. Removing the relation-aware encoder (w/o RA) causes the most severe performance degradation across all datasets, demonstrating its essential role in decomposing heterogeneous graphs into bipartite subgraphs for fine-grained semantic extraction. While meta-graph encoder ablation (w/o MG) also reduces performance, its consistently smaller impact versus w/o RA reveals it is less effective than the relation-aware encoder due to its lack of fine-grained semantic extraction capabilities. Contrastive learning removal (w/o Contrast) induces cross-dataset declines, most acutely on IMDB, demonstrating its necessity for aligning dual-channel feature spaces and mitigating semantic confusion in complex multi-relational contexts. Finally, eliminating the gated interaction module (w/o Interaction) significantly compromises performance, particularly on the ACM and IMDB datasets, validating its core function in progressively integrating cross-channel information to derive consensus representations.

To validate module efficacy beyond classification tasks, we conduct node clustering ablation studies on DBLP and ACM datasets (Table 6). Consistent with classification results, removing any module induces notable performance degradation. This cross-task robustness demonstrates that each module provides non-redundant contributions: RA enables semantic decomposition through bipartite subgraphs, MG supplements higher-order structural priors, contrastive learning aligns feature spaces across channels and gated interaction synthesizes consensus representations. Collectively, they form an interdependent framework essential for advanced heterogeneous graph representation learning.

To further validate the effect of node feature-based aggregation coefficients and topology-based aggregation coefficients on the model, we perform the corresponding performance comparison. The results are

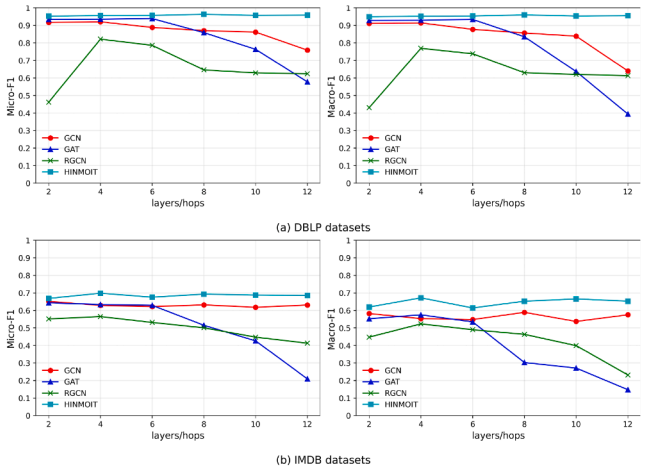


Fig. 4. Quantitative performance analysis of different methods with different layers/hops settings on different datasets.

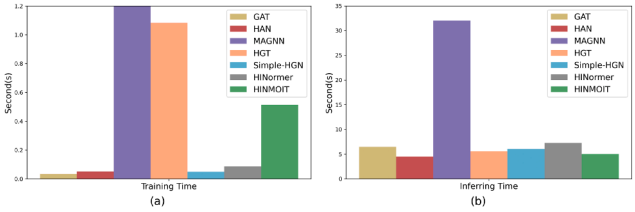


Fig. 5. Scalability study. (a) Training Time. (b) Inferring Time.

shown in Table 7. Here, alpha represents the topology-based aggregation coefficient. In the RA module, it is defined by the degree between nodes, indicating the relative importance of different neighbors. In the MG module, it is measured by the number of edges between nodes within the meta-graph, which similarly reflects different levels of heterogeneous similarity present in the heterogeneous graph. Beta represents the attention coefficient generated based on the content features of the nodes. From the table, we observe that removing the beta coefficient in the meta-graph encoder module leads to significant performance degradation on the DBLP and ACM datasets. This result indicates that the attention mechanism based on node content features within the meta-graph encoder plays a crucial role in enhancing the performance of HINMOIT. The removal of other modules also results in varying degrees of performance degradation. However, on the IMDB dataset eliminating the alpha coefficient in the meta-graph module slightly improves the Micro-F1 score, suggesting potential conflicts between different modules in this dataset. These results demonstrate that the two coefficients are important for aggregating the structural and semantic information.

5.5. Resistance to over-Smoothing

To further examine HINMOIT's ability to preserve discriminative representations under deep propagation where over-smoothing typically becomes severe, we conduct a depth-sensitivity study. HINMOIT adopts a dual-channel encoder architecture with a gated interaction mechanism that progressively aligns and fuses multi-order features, aiming to mitigate the feature over-smoothing problem when aggregating multi-hop neighbors.

Specifically, we systematically vary the model depth from 2 to 12 and benchmark HINMOIT against representative baselines (GCN, GAT, and RGCN) on the DBLP and IMDB datasets, reporting both Micro-F1 and Macro-F1 (Fig. 4). On DBLP, GAT exhibits pronounced performance degradation as depth exceeds 8 layers, while GCN begins to decline noticeably beyond 10 layers, reflecting the classic signature of over-smoothing where node representations progressively lose discriminabil-

Table 5
Ablation study on node classification.

Dataset	DBLP		ACM		IMDB	
Model \ Metrics	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
w/o RA	92.11±0.50	91.59±0.51	90.89±0.87	90.97±0.84	67.39±0.20	64.29±0.45
w/o MG	95.60±0.05	95.27±0.08	93.25±0.33	93.30±0.34	68.00±0.14	66.31±0.09
w/o Contrast	95.53±0.15	95.19±0.16	93.34±0.33	93.40±0.31	68.00±0.28	62.43±0.49
w/o Interaction	95.21±0.10	94.85±0.09	92.97±0.20	93.03±0.21	68.67±0.22	63.66±0.12
HINMOIT	**95.85±0.27**	**95.49±0.22**	**94.12±0.43**	**94.19±0.23**	**69.75±0.47**	**67.56±0.34**

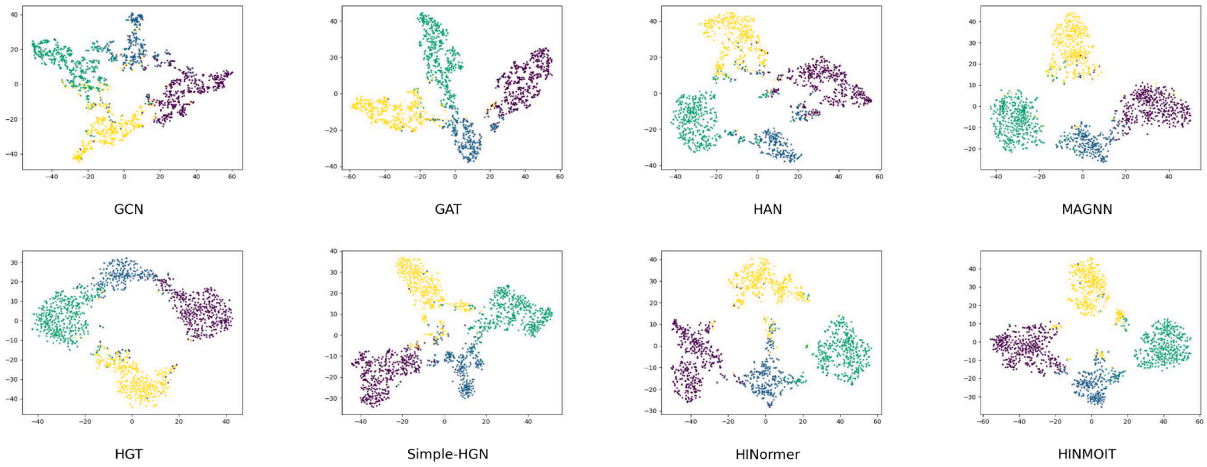


Fig. 6. Visualization of node representations on DBLP.

Table 6
Ablation study on node clustering.

Dataset	DBLP		ACM	
Model \ Metrics	NMI	ARI	NMI	ARI
w/o RA	75.26	80.25	67.38	72.86
w/o MG	84.34	88.93	74.03	79.65
w/o Contrast	83.43	88.01	74.23	80.39
w/o Interaction	83.72	88.40	72.84	79.07
HINMOIT	**85.25**	**89.68**	**77.74**	**83.49**

ity. On IMDB, the depth sensitivity becomes more severe: when depth reaches around 6, GAT and RGCN show clear performance decreases, and the degradation further intensifies at deeper settings. In contrast, HINMOIT remains stable across the entire depth range, maintaining strong performance even at 10–12 layers on both datasets.

These results demonstrate that HINMOIT can exploit larger receptive fields without suffering the dramatic depth-induced performance collapse observed in standard message-passing baselines. The robustness is attributable to its dual-channel and multi-scale representation learning, which simultaneously captures fine-grained local patterns and broader semantic dependencies, thereby preserving discriminative node embeddings under deep propagation.

5.6. Complexity analysis

To validate the computational efficiency of HINMOIT, we conduct a comparative analysis of training time per epoch and inference time against six representative baselines, including GAT, HAN, MAGNN, HGT, Simple-HGN, and HINormer. As illustrated in Fig. 5, HINMOIT exhibits a favorable balance between efficiency and performance. It achieves lower training costs compared to complex models like MAGNN and HGT, while delivering better performance. Moreover, its inference time is shorter than that of most baseline models, underscoring the efficiency of our proposed architecture.

Further, we provide a formal derivation of the time and space complexity. Let $N = |V|$ be the number of nodes, $M = |E|$ the number of edges, and $R = |\mathcal{R}|$ the number of relation types in the original HIN. Let $M_{mg} = |E_{mg}|$ denote the number of edges in the constructed meta-graph. We use d for the hidden dimension, K for the number of attention heads, and L_{ra}/L_{mg} for the numbers of layers in the RA/MG encoders.

Time Complexity. For HINMOIT, the runtime is dominated by sparse edge-wise message passing. Relation-aware encoder aggregates neighbors based on relation-specific bipartite graphs. The complexity is $O(L_{ra}KMd)$. Meta-graph encoder module performs aggregation over the meta-graph structure and the original 1-hop neighborhood. The cost is $O(L_{mg}K(M + M_{mg})d)$. The node-wise gating and residual fusion involve element-wise operations with a complexity of $O(Nd)$, which is negligible compared to the aggregation steps. Consequently, the total time complexity is $T_{\text{HINMOIT}} = O\big(Kd\big((L_{ra} + L_{mg})M + L_{mg}M_{mg}\big)\big)$. This indicates that HINMOIT scales linearly with the graph size M, avoiding quadratic costs associated with dense mechanisms.

Space Complexity. Memory complexity is dominated by node embeddings $O(Nd)$ and sparse storage over processed edges $O\big(K(M + M_{mg})\big)$. Thus, $S_{\text{HINMOIT}} = O\big(Nd + K(M + M_{mg})\big) + S_{\text{param}}$, where $S_{\text{param}} = O(L_{ra}KRd^2)$ with relation-specific projections.

Compared with the time complexity of MAGNN and HGT, MAGNN incurs instance-level meta-path encoding, with M_p instances per node and length ℓ, $T_{\text{MAGNN}} = O(NM_p\ell d)$ (up to attention and head factors). In contrast, HGT is edge-linear with $T_{\text{HGT}} = O(LKMd + LNd^2)$, where the second term accounts for type-specific linear projections. Therefore, HINMOIT eliminates the instance-dependent $O(NM_p\ell d)$ cost relative to MAGNN and preserves edge-linear scaling comparable to HGT, while adding $O(L_{mg}KM_{mg}d)$ due to meta-graph aggregation, which is bounded by sparse construction.

For large-scale settings with $N > 10^6$ or $R > 100$, HINMOIT remains robust. Its edge-linear nature ensures feasible runtimes on large graphs. For $N > 10^6$, full-batch training is often memory-bound, and mini-batch neighbor sampling reduces activation storage from $O(Nd)$ to $O(Bd)$. For $R > 100$, the parameter term can be controlled by sharing or factorizing relation-specific transforms.

Table 7
Impact of different aggregation coefficients on node classification.

Dataset	DBLP		ACM		IMDB	
Model \ Metrics	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
w/o RA-alpha	95.49	95.16	93.30	93.34	68.93	66.20
w/o RA-beta	95.49	95.13	93.30	93.35	69.63	66.79
w/o MG-alpha	95.42	95.09	93.39	93.45	**69.95**	66.76
w/o MG-beta	95.14	94.74	92.82	92.88	68.90	65.66
w/o RA,MG-alpha	95.35	95.03	93.39	93.48	69.85	66.59
w/o RA,MG-beta	95.49	95.20	93.01	93.07	69.59	66.23
HINMOIT	**95.92**	**95.59**	**94.12**	**94.19**	69.75	**67.56**

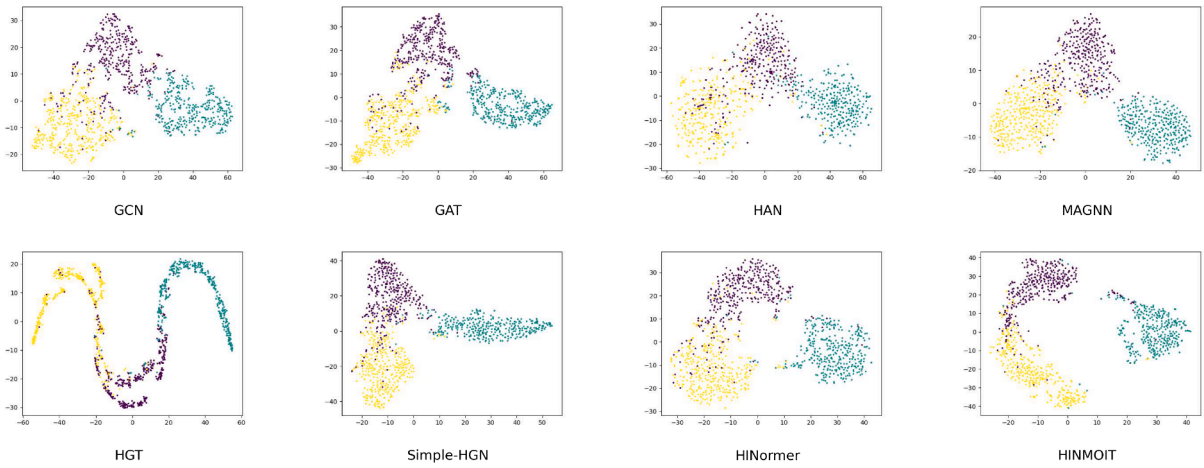


Fig. 7. Visualization of node representations on ACM.

5.7. Visualization

To intuitively demonstrate the quality of the learned node representations, we project the low-dimensional embeddings into a two-dimensional space using t-SNE and visualize the clustering results on the DBLP and ACM datasets. As shown in Fig. 6 and Fig. 7, different colors represent different categories. Compared with the baseline model, HINMOIT produces more distinct and compact clusters with larger inter-cluster distances, indicating clearer category boundaries and better semantic separation. These results further validate the effectiveness of HINMOIT in learning heterogeneous graph embeddings for the node clustering task.

6. Conclusion

In this study, we propose HINMOIT, a novel representation learning method for heterogeneous information networks based on multi-order information extraction and interaction. For multi-order information extraction, we introduce dual encoders to learn both local and global heterogeneous graph information. The low-order relation-aware encoder captures local structure using residual connections to preserve node information across layers. In contrast, the high-order meta-graph encoder extracts high-level semantics through 1-hop aggregation. The two encoders complement each other to further promote the mining of heterogeneity, solving the over-smoothing problem of only using low-order information, and the fine-grained semantic loss problem of only using high-order information. For multi-order information interaction, we introduce a gated information interaction module to dynamically align features of the dual encoders within a unified space, resolving semantic inconsistencies and enabling complementary information fusion. Extensive validation on benchmark datasets demonstrates that HINMOIT outperforms state-of-the-art methods in node classification and clustering tasks.

CRediT authorship contribution statement

Yanglan Gan: Investigation, Conceptualization, Writing – review & editing, Methodology; **Lideng Cai:** Writing – original draft, Writing – review & editing, Conceptualization, Methodology, Formal analysis, Data curation; **Kaili Wang:** Investigation,; **Guangwei Xu:** Investigation; **Guobing Zou:** Investigation, Writing – review & editing.

Data availability

Data will be made available on request.

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.

Acknowledgments

This work was sponsored in part by the National Natural Science Foundation of China (62572114, 62272290 and 62477006).

References

[1] L. Weng, Q. Zhang, Z. Lin, L. Wu, Harnessing heterogeneous social networks for better recommendations: a grey relational analysis approach, Expert Syst. Appl. 174 (2021) 114771.
[2] A.C.M. da Silva, D.F. Silva, R.M. Marcacini, Multimodal representation learning over heterogeneous networks for tag-based music retrieval, Expert Syst. Appl. 207 (2022) 117969.
[3] E.F. Maleki, N. Ghadiri, M.L. Shahreza, Z. Maleki, DHLP 1&2: Giraph based distributed label propagation algorithms on heterogeneous drug-related networks, Expert Syst. Appl. 159 (2020) 113640.

[4] X. Wang, X. He, Y. Cao, M. Liu, T.-S. Chua, Kgat: knowledge graph attention network for recommendation, in: Proceedings of the 25Th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 950–958.
[5] C. Shi, Y. Li, J. Zhang, Y. Sun, P.S. Yu, A survey of heterogeneous information network analysis, IEEE Trans. Knowl. Data Eng. 29 (1) (2016) 17–37.
[6] Y. Sun, J. Han, X. Yan, P.S. Yu, T. Wu, Pathsim: meta path-based top-k similarity search in heterogeneous information networks, Proc. VLDB Endowment 4 (11) (2011) 992–1003.
[7] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P.S. Yu, A comprehensive survey on graph neural networks, IEEE Trans. Neural Netw. Learn. Syst. 32 (1) (2020) 4–24.
[8] D. Luo, W. Cheng, W. Yu, B. Zong, J. Ni, H. Chen, X. Zhang, Learning to drop: robust graph neural network via topological denoising, in: Proceedings of the 14Th ACM International Conference on Web Search and Data Mining, 2021, pp. 779–787.
[9] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, P.S. Yu, Heterogeneous graph attention network, in: The World Wide Web Conference, 2019, pp. 2022–2032.
[10] Q. Lv, M. Ding, Q. Liu, Y. Chen, W. Feng, S. He, C. Zhou, J. Jiang, Y. Dong, J. Tang, Are we really making much progress? revisiting, benchmarking and refining heterogeneous graph neural networks, in: Proceedings of the 27Th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 1150–1160.
[11] Z. Hu, Y. Dong, K. Wang, Y. Sun, Heterogeneous graph transformer, in: Proceedings of the Web Conference 2020, 2020, pp. 2704–2710.
[12] G. Zhu, Z. Zhu, W. Wang, Z. Xu, C. Yuan, Y. Huang, Autoac: towards automated attribute completion for heterogeneous graph neural network, in: 2023 IEEE 39Th International Conference on Data Engineering (ICDE), IEEE, 2023, pp. 2808–2821.
[13] L. Yu, L. Sun, B. Du, C. Liu, W. Lv, H. Xiong, Heterogeneous graph representation learning with relation awareness, IEEE Trans. Knowl. Data Eng. 35 (6) (2022) 5935–5947.
[14] J. Liu, L. Song, G. Wang, X. Shang, Meta-hgt: metapath-aware hypergraph transformer for heterogeneous information network embedding, Neural Netw. 157 (2023) 65–76.
[15] X. Yang, M. Yan, S. Pan, X. Ye, D. Fan, Simple and efficient heterogeneous graph neural network, in: Proceedings of the AAAI Conference on Artificial Intelligence, 37, 2023, pp. 10816–10824.
[16] M. Guan, X. Cai, J. Shang, F. Hao, D. Liu, et al., HMSG: Heterogeneous graph neural network based on metapath subgraph learning, Knowl. Based Syst. 279 (2023) 110930. https://doi.org/10.1016/j.knosys.2023.110930
[17] W. Sun, Y. Cheng, X. Zhao, HGNN-NHSA: Heterogeneous graph neural network based on neighbor and high-order subgraph aggregation learning, Neurocomputing (2025) 129618.
[18] L. Yang, F. Wu, Z. Zheng, B. Niu, J. Gu, C. Wang, X. Cao, Y. Guo, Heterogeneous graph information bottleneck, in: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI), 2021.
[19] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, Q. Mei, Line: large-scale information network embedding, in: Proceedings of the 24Th International Conference on World Wide Web, 2015, pp. 1067–1077.
[20] B. Perozzi, R. Al-Rfou, S. Skiena, Deepwalk: online learning of social representations, in: Proceedings of the 20Th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014, pp. 701–710.
[21] A. Grover, J. Leskovec, Node2vec: scalable feature learning for networks, in: Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 855–864.
[22] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks,(2017), arXiv:1710.10903.
[23] W. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, Adv. Neural Inf. Process. Syst. 30 (2017).
[24] K. Xu, W. Hu, J. Leskovec, S. Jegelka, How powerful are graph neural networks?,(2018), arXiv:1810.00826.
[25] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks,(2016), arXiv:1609.02907.
[26] W.-L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, C.-J. Hsieh, Cluster-gcn: an efficient algorithm for training deep and large graph convolutional networks, in: Proceedings of the 25Th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 257–266.
[27] Y. Rong, W. Huang, T. Xu, J. Huang, Dropedge: towards deep graph convolutional networks on node classification,(2019), arXiv:1907.10903.
[28] Z. Ying, D. Bourgeois, J. You, M. Zitnik, J. Leskovec, Gnnexplainer: generating explanations for graph neural networks, Adv. Neural Inf. Process. Syst. 32 (2019).
[29] D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen, X. Zhang, Parameterized explainer for graph neural network, Adv. Neural Inf. Process. Syst. 33 (2020) 19620–19631.
[30] P. Veličković, W. Fedus, W.L. Hamilton, P. Liò, Y. Bengio, R.D. Hjelm, Deep graph infomax, arXiv:1809.10341 (2018).
[31] H. Hafidi, M. Ghogho, P. Ciblat, A. Swami, Graphcl: contrastive self-supervised learning of graph representations,(2020), arXiv:2007.08025.
[32] Z. Peng, W. Huang, M. Luo, Q. Zheng, Y. Rong, T. Xu, J. Huang, Graph representation learning via graphical mutual information maximization, in: Proceedings of the Web Conference 2020, 2020, pp. 259–270.
[33] Z. Li, X. Wang, J. Zhao, W. Guo, J. Li, Hycube: efficient knowledge hypergraph 3D circular convolutional embedding, IEEE Trans. Knowl. Data Eng. 37 (4) (2025) 1902–1914. https://doi.org/10.1109/TKDE.2025.3531372
[34] Z. Li, C. Wang, X. Wang, Z. Chen, J. Li, HJE: Joint convolutional representation learning for knowledge hypergraph completion, IEEE Trans. Knowl. Data Eng. 36 (8) (2024) 3879–3892. https://doi.org/10.1109/TKDE.2024.3365727
[35] W. Guo, Z. Li, X. Wang, Z. Chen, J. Zhao, J. Li, Y. Yuan, Convd: attention enhanced dynamic convolutional embeddings for knowledge graph completion, IEEE Trans. Knowl. Data Eng. 37 (9) (2025) 5049–5062. https://doi.org/10.1109/TKDE.2025.3582243
[36] Y. Dong, N.V. Chawla, A. Swami, Metapath2vec: scalable representation learning for heterogeneous networks, in: Proceedings of the 23Rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 135–144.
[37] T.-y. Fu, W.-C. Lee, Z. Lei, Hin2vec: explore meta-paths in heterogeneous information networks for representation learning, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1797–1806.
[38] X. Fu, J. Zhang, Z. Meng, I. King, Magnn: metapath aggregated graph neural network for heterogeneous graph embedding, in: Proceedings of the Web Conference 2020, 2020, pp. 2331–2341.
[39] S. Yun, M. Jeong, R. Kim, J. Kang, H.J. Kim, Graph transformer networks, Adv. Neural Inf. Process. Syst. 32 (2019).
[40] C. Wang, S. Zhou, K. Yu, D. Chen, B. Li, Y. Feng, C. Chen, Collaborative knowledge distillation for heterogeneous information network embedding, in: Proceedings of the ACM Web Conference 2022, 2022, pp. 1631–1639.
[41] M. Schlichtkrull, T.N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15, Springer, 2018, pp. 593–607.
[42] S. Zhu, C. Zhou, S. Pan, X. Zhu, B. Wang, Relation structure-aware heterogeneous graph neural network, in: 2019 IEEE International Conference on Data Mining (ICDM), IEEE, 2019, pp. 1534–1539.
[43] C. Zhang, D. Song, C. Huang, A. Swami, N.V. Chawla, Heterogeneous graph neural network, in: Proceedings of the 25Th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 793–803.
[44] Q. Mao, Z. Liu, C. Liu, J. Sun, Hinormer: representation learning on heterogeneous information networks with graph transformer, in: Proceedings of the ACM Web Conference 2023, 2023, pp. 599–610.
[45] Z. Zhou, J. Shi, R. Yang, Y. Zou, Q. Li, SlotGAT: slot-based message passing for heterogeneous graphs, in: International Conference on Machine Learning, PMLR, 2023, pp. 42644–42657.
[46] X. Wang, N. Liu, H. Han, C. Shi, Self-supervised heterogeneous graph neural network with co-contrastive learning, in: Proceedings of the 27Th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 1726–1736.
[47] H. Hong, H. Guo, Y. Lin, X. Yang, Z. Li, J. Ye, An attention-based graph neural network for heterogeneous structural learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 34, 2020, pp. 4132–4139.
[48] Y. Tang, Y. Huang, J. Hou, Z. Liu, Type-adaptive graph transformer for heterogeneous information networks, Appl. Intell. 54 (22) (2024) 11496–11509.
[49] J. Yuan, S. Lu, P. Duan, J. He, AGHINT: Attribute-guided representation learning on heterogeneous information networks with transformer, Knowl. Based Syst. 310 (2025) 112977.