Structure-Enhanced Heterogeneous Graph Contrastive Learning

Yanqiao Zhu[*†‡] Yichen Xu[*§] Hejie Cui[¶] Carl Yang[¶] Qiang Liu[†‡] Shu Wu[†‡‖]

Abstract

Recent years have seen a growing interest in Graph Contrastive Learning (GCL), which trains Graph Neural Network (GNN) model to discriminate similar and dissimilar pairs of nodes without human annotations. Most prior GCL work focuses on homogeneous graphs and little attention has been paid to Heterogeneous Graphs (HGs) that involve different types of nodes and edges. Moreover, earlier studies reveal that the explicit use of structure information of underlying graphs is useful for learning representations. Conventional GCL methods merely measure the likelihood of contrastive pairs according to node representations, which may not align with the true semantic similarities. How to leverage such structure information for GCL is not yet well-understood. To address the aforementioned challenges, this paper presents a novel method dubbed STructure-EnhaNced heterogeneous graph ContrastIve Learning, STENCIL for brevity. At first, we generate multiple semantic views for HGs based on metapaths. Unlike most methods that maximize the consistency among these views, we propose a novel multiview contrastive aggregation objective to adaptively distill information from each view. In addition, we advocate the explicit use of structure embedding, which enriches the model with local structural patterns of the underlying HGs, so as to better mine true and hard negatives for GCL. Empirical studies on three real-world datasets show that our proposed method consistently outperforms existing state-of-the-art methods and even surpasses several supervised counterparts.

1 Introduction

Many real-world complex interactive objectives can be represented in Heterogeneous Graphs (HGs) or heterogeneous information networks. Recent development in heterogeneous Graph Neural Networks (GNNs) has achieved great success in analyzing heterogenous data [27, 29]. However, most existing models require a large amount of labeled data for proper training [7, 14, 35, 36], which may not be accessible in reality. As a promising strategy of leveraging abundant unlabeled data, Graph Contrastive Learning (GCL) is proposed to learn representations by distinguishing semantically similar samples (positives) over dissimilar samples (negatives) in the latent space without human annotations. Most existing GCL models follow a multiview paradigm [35, 46, 47], where multiple views of the input data are constructed via stochastic augmentations and the model is then trained to maximize the consistency of representations among these views.

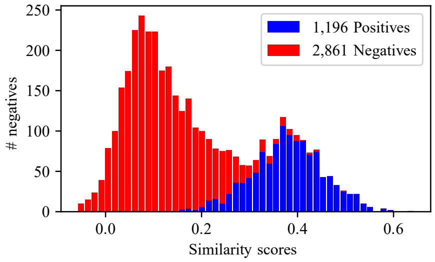


Figure 1: A histogram of negatives and their semantic similarity scores with an arbitrary anchor node. With the similarity to the anchor node increasing, there are more positive samples (false negatives), indicating a mismatch with the true semantic relationship.

Though multiview GCL has achieved promising performance in many tasks, it is still non-trivial to adopt multiview GCL on HG data. In HGs, multiple types of nodes and edges convey rich semantic information. It is therefore natural to construct multiple views based on metapaths. Then, following the multiview contrastive objective [46, 47], its embeddings in different views constitute positives and all other embeddings are regarded as negative examples. Nevertheless, this scheme fails to consider the inter-view dependency of different views (e.g., complementary or redundant information [28]) and may lead to suboptimal performance. For example, consider an academic network, where nodes correspond to four types of entities: Papers (P), Conferences (C), Topics (T), and Authors (A). Two views created by APA and APCPA share *common* co-authorship information, while two other metapaths APCPA and APTPA connect authors from two *dissimilar* sources: conferences and topics. Therefore, it is insufficient to distill comprehensive information from HGs by only contrasting node representations within each metapath-induced view.

Furthermore, conventional GCL methods usually ran-

[*] The first two authors contributed equally to this research.

[†] Center for Research on Intelligent Perception and Computing, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

[‡] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China.

[§] School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, China.

[¶] Department of Computer Science, Emory University, GA, USA.

[‖] To whom correspondence should be addressed (shu.wu@nlpr.ia.ac.cn).

domly select negative samples from other nodes and measure the likelihood of contrastive pairs merely using node representations, which may not well align with the true semantic relationship [40, 45]. To see this phenomenon clearly, we conduct an oracle-based analysis on DBLP, a widely-used academic network. Specifically, we plot the relationship between negatives and their similarity scores with one arbitrary anchor node. As shown in Figure 1, with the similarity of negatives to the anchor increasing, there are more *false* negatives samples (i.e. nodes sharing the same label with the anchor). A possible explanation is that, when neighborhood aggregation is performed in each semantic view [36], a heterogeneous GNN produces similar embeddings within ego networks. Embeddings of neighboring nodes sharing *the same label* with the anchor node thus tend to be similar to the anchor. In addition, at the beginning of training, node embeddings are suffered from poor quality, which may be another obstacle of sampling true negatives.

The above discussion motivates us to quantify the likelihood of contrastive pairs from structural aspects. Previous work has established that explicitly incorporating structure information of underlying graphs is beneficial for learning representations [3, 5, 15, 16, 42]. For sampling negative instances in GCL, we argue that it is not only important to consider *individual* node representations but also *local* structural representations. These structural embeddings enhance the graph model by encoding additional graph-based closeness between nodes and thus carry local positional information and context of the given graph-structured data.

In this paper, we propose STructure-EnhaNced heterogeneous graph ContrastIve Learning, STENCIL for brevity. At first, our model works by constructing multiple views corresponding to metapaths and obtaining node embeddings within each view through a heterogeneous GNN. Then, we propose a novel multiview contrastive aggregation objective for HG data, whose aim is to ensure global consistency among metapath-induced views and adaptively encode information from each view. Thereafter, we propose to digest *structural characteristics* of underlying HGs rather than simply assessing similarity using node embeddings. In particular, we measure the similarity of each negative pair according to structure embeddings. Then, we select negatives with largest similarities and synthesize more negatives by randomly mixing up these selected negatives. In this way, these synthesized samples upweight true and hard negative samples from both semantic and structural aspects of HGs. Our structure-enhanced GCL scheme enjoys another benefit of being irrespective of the training progress, which could improve the selection of negatives even in the initial training stage.

In summary, the main contribution of this work is threefold:

- We propose a novel STENCIL model that enables self-supervised training for HGs. Specifically, we propose a novel contrastive aggregation objective that adaptively learn information from each semantic view.
- To further improve the performance of negative sampling in GCL for HGs, we propose to enrich the model with structurally hard negatives.
- Extensive experiments on three real-word datasets from various domains demonstrate the effectiveness of the proposed method. Particularly, our STENCIL method outperforms representative unsupervised baselines, achieves competitive performance with supervised counterparts, and even exceeds several of them.

To foster reproducible research, we make all the code publicly available at https://github.com/CRIPAC-DIG/STENCIL.

2 Preliminaries

2.1 Problem Definition. We introduce several key definitions of heterogeneous graphs and the problem of unsupervised heterogeneous graph representation learning.

DEFINITION 2.1. *A Heterogeneous Graph (HG), denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{X}, \boldsymbol{R}, \phi, \varphi)$, is a graph with multiple types of nodes and edges, where $\mathcal{V}, \mathcal{E}$ denote the node set and the edge set respectively. The node type mapping function $\phi : \mathcal{V} \to \mathcal{S}$ associates each node $v_i \in \mathcal{V}$ with a node type $s = \phi(v_i)$, the edge type mapping function $\varphi : \mathcal{E} \to \mathcal{R}$ associates each edge $e_{ij} \in \mathcal{E}$ with an edge type $r = \varphi(e_{ij})$, with $|\mathcal{S}| + |\mathcal{R}| > 2$. Moreover, each node v_i and each edge e_{ij} is possibly associated with attribute $\boldsymbol{x}_i$ and $\boldsymbol{r}_{ij}$. Note that the edge type $r = \varphi(e_{ij})$ implicitly defines types of its two end nodes v_i and v_j.*

DEFINITION 2.2. *A metapath p defines a path on the network schema in the form of $s_1 \xrightarrow{r_1} s_2 \xrightarrow{r_2} \cdots \xrightarrow{r_l} s_{l+1}$. It represents a composite relation $r_1 \circ r_2 \circ \cdots \circ r_l$ between two nodes v_1 and v_{l+1} that captures the proximity between the two nodes from a particular perspective, where $\circ$ is the composite operator. We further denote the set of all considered metapaths as $\mathcal{P}$.*

DEFINITION 2.3. *Given a HG $\mathcal{G}$, the problem of heterogeneous graph representation learning aims to learn node representations $\boldsymbol{H} \in \mathbb{R}^{|\mathcal{V}| \times d}$ that encode both structural and semantic information, where $d \ll |\mathcal{V}|$ is the dimension of the embedding space.*

2.2 Heterogeneous Graph Neural Networks. Most heterogeneous GNNs [7, 36] learn node representations under different views and then aggregate them using attention networks. Following their approaches, we first generate multiple views, each corresponding to one metapath that encodes one aspect of information. Then, we leverage an attentive network to compute metapath-specific embedding $\boldsymbol{h}_i^p$ for node

v_i under metapath p as

$$\text{(2.1)} \qquad \boldsymbol{h}_i^p = \Big\Vert_{k=1}^{K} \sigma\left(\sum_{v_j \in \mathcal{N}_p(v_i)} \alpha_{ij}^p \boldsymbol{W}^p \boldsymbol{x}_j\right),$$

where $\Vert$ concatenates K standalone node representations in each attention head, $\mathcal{N}_r(v_i)$ defines the neighborhood of v_i that is connected by metapath p, $\boldsymbol{W}^p \in \mathbb{R}^{d\times m}$ is a linear transformation matrix for metapath p, and $\sigma(\cdot)$ is the activation function, such as $\mathrm{ReLU}(\cdot) = \max(0, \cdot)$. The attention coefficient α_{ij}^p can be computed by a softmax function

$$\text{(2.2)} \qquad \alpha_{ij}^p = \frac{\exp(\sigma(\boldsymbol{a}_p^\top [\boldsymbol{h}_i^p \parallel \boldsymbol{h}_j^p]))}{\sum_{v_k \in \mathcal{N}_p(v_i)} \exp(\sigma(\boldsymbol{a}_p^\top [\boldsymbol{h}_i^p \parallel \boldsymbol{h}_k^p]))},$$

where $\boldsymbol{a}_p \in \mathbb{R}^{2d}$ is a trainable metapath-specific linear weight vector.

Finally, we combine node representation in each view to an aggregated representation. We employ another attentive network to obtain the aggregated representation $\boldsymbol{h}_i$ that combines information from every semantic space by

$$\text{(2.3)} \qquad \boldsymbol{h}_i = \sum_{p=1}^{|\mathcal{P}|} \beta^p \boldsymbol{h}_i^p.$$

The coefficients are given by

$$\text{(2.4)} \qquad \beta^p = \frac{\exp(w^p)}{\sum_{p' \in \mathcal{P}} \exp(w^{p'})},$$

$$\text{(2.5)} \qquad w^p = \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \boldsymbol{q}^\top \cdot \tanh(\boldsymbol{W}\boldsymbol{h}_i^p + \boldsymbol{b}),$$

where $\boldsymbol{q} \in \mathbb{R}^{d_m}$ is the aggregation attention vector, $\boldsymbol{W} \in \mathbb{R}^{d_m \times d}$, $\boldsymbol{b} \in \mathbb{R}^{d_m}$ is the weight matrix and the bias vector respectively, and $d_m \in \mathbb{R}$ is a hyperparameter.

3 The Proposed Method: STENCIL

In the following section, we present the proposed STENCIL in detail. There are three major components in the proposed STENCIL framework: (a) a heterogeneous graph encoder, which embeds nodes under each metapath-induced view into low-dimensional vectors and aggregates these embeddings into a final representation, (b) a multiview contrastive aggregation objective that adaptively encodes node representations in a self-supervised manner, and (c) structure-enriched negative mining, which discovers and reweights structurally hard samples.

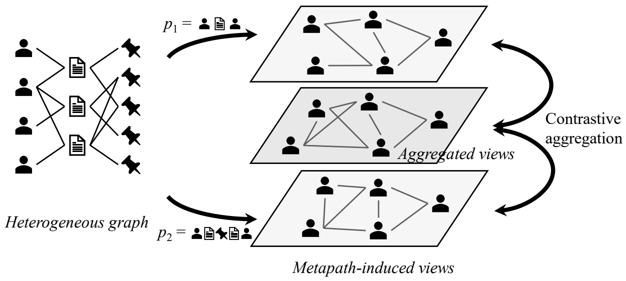


Figure 2: Illustrating the proposed multiview contrastive aggregation scheme. We construct multiple views induced by metapaths and learn representations with heterogeneous GNNs. Then, we train the model with a multiview contrastive aggregation objective that adaptively distills essential information from each view.

3.1 Multiview Contrastive Aggregation. As described in Section 2.2, we first generate multiple views according to metapaths and learn node representations under these views independently using heterogeneous GNN models. Then, to comprehensively learn the information encoded in different metapaths, we propose a novel multiview contrastive aggregation objective, which aims to *maximize the agreement between the node representation under a specific metapath view and an aggregated representation for all metapaths*. This contrastive aggregation scheme is shown in Figure 2. Its learning objective can be mathematically expressed as

$$\text{(3.6)} \qquad \max \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \left[\frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \frac{1}{2}\left(I(\boldsymbol{h}_i^p; \boldsymbol{h}_i) + I(\boldsymbol{h}_i; \boldsymbol{h}_i^p)\right)\right],$$

where $\boldsymbol{h}_i^p$ is a metapath-specific embedding for node v_i under metapath p and $\boldsymbol{h}_i$ is the aggregated embedding for node v_i that collects information of all its relations.

Following previous work [10, 32, 45], to estimate the mutual information $I(\boldsymbol{h}_i^p; \boldsymbol{h}_i)$ in Eq. (3.6), we empirically choose the InfoNCE estimator. Concretely, for node representation $\boldsymbol{h}_i^p$ in one specific metapath-induced view, we construct its positive sample as the aggregated representation $\boldsymbol{h}_i$, while embeddings of all other nodes in the semantic and the aggregated embeddings are considered as negative samples. The contrastive loss can be expressed by

$$\text{(3.7)} \qquad \mathcal{L}(\boldsymbol{h}_i^p, \boldsymbol{h}_i) = -\log \frac{e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h}_i)/\tau}}{e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h}_i)/\tau} + \sum\limits_{j \neq i}\left(e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h}_j)/\tau} + e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h}_j^p)/\tau}\right)},$$

where $\tau \in \mathbb{R}$ is a temperature parameter. We define the critic function $\theta(\cdot, \cdot)$ by

$$\theta(\boldsymbol{h}_i, \boldsymbol{h}_j) = \frac{g(\boldsymbol{h}_i)^\top g(\boldsymbol{h}_j)}{\|g(\boldsymbol{h}_i)\|\|g(\boldsymbol{h}_i)\|},$$

where $g(\cdot)$ is parameterized by a non-linear multilayer perceptron to enhance the expressive power [2].

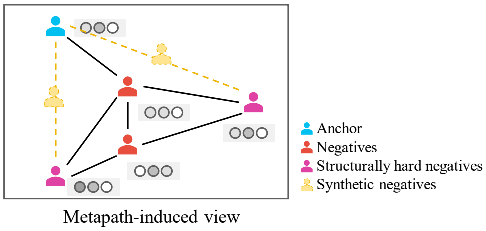


Figure 3: The proposed structure-enhanced negative mining scheme, which discovers hard negatives in each metapath-induced view using structure embeddings.

3.2 Structure Embeddings for Metapath-Induced Views.

In the context of HGs, we observe that metapath-level node representations are not sufficient to calculate the similarity of each negative pair. Therefore, in this work, to effectively measure the likelihood of each negative sample with respect to the anchor, we propose to digest the structural characteristics of the underlying graphs.

In this paper, we propose two model variants STENCIL-PPR and STENCIL-PE, which use Laplacian positional embeddings and personalized PageRank scores for modeling local structural patterns of the metapath-induced view, respectively. Note that in order to empower the model with inductive capabilities, we prefer a local measure to a global one. Specifically, we introduce a structural metric $s(i, j, p)$ representing the distance measure of a negative node v_i to the anchor v_j given structural embeddings in metapath-induced view p.

- The Personalized PageRank (PPR) score [12, 18] of node v is defined as the stationary distribution of a random walk starting from and returning to node v at a probability of c at each step. Formally, the PPR vector of node v under semantic view p satisfies the following equation

  $$\boldsymbol{s}_v^p = (1 - c)\boldsymbol{A}^p \boldsymbol{s}_v^p + c\boldsymbol{I}\boldsymbol{p}_v, \tag{3.8}$$

  where c is the returning probability and $\boldsymbol{p}_v$ is the preference vector with $(\boldsymbol{p}_v)_i = 1$ when $i = v$ and all other entries set to 0. $\boldsymbol{A}^p$ denotes the adjacency matrix induced by metapath p. The structural similarity $s(v, k, p)$ between node v and k can be represented by the PPR score of node k with respect to v, i.e. $(\boldsymbol{s}_v^p)_k$.
- The Laplacian positional embedding of one node is defined to be its k smallest non-trivial eigenvectors [5]. We simply define the structure similarity $s(v, k, p)$ as the inner product between $\boldsymbol{s}_v^p$ and $\boldsymbol{s}_k^p$.

3.3 Structure-Enhanced Negative Mining.

Previous studies [1, 25, 41] demonstrate that CL benefits from hard negative samples, i.e. samples close to the anchor node such that cannot be distinguished easily. As illustrated in Figure 3, after obtaining structural embeddings for each metapath-induced view, we perform negative mining by giving larger weights to structurally harder negative samples. To be specific, we sort negatives according to the hardness metric $s(i, j, p)$ and pick the top-T negatives to form a candidate list for metapath-induced view p. Then, we synthesize $M \ll |\mathcal{V}|$ samples by creating a convex linear combination of them. The generated sample $\widetilde{\boldsymbol{h}}_m^p$ can be written as

$$\widetilde{\boldsymbol{h}}_m^p = \alpha_m \boldsymbol{h}_i^p + (1 - \alpha_m)\boldsymbol{h}_j^p, \tag{3.9}$$

where $\boldsymbol{h}_i^p, \boldsymbol{h}_j^p \in \mathcal{B}^p$ are randomly picked from the memory bank, $\alpha_m \sim \mathrm{Beta}(\alpha, \alpha)$, and $\alpha \in \mathbb{R}$ is a hyperparameter, fixed to 1 in our experiments. These interpolated samples will be added into negative bank when estimating mutual information $I(\boldsymbol{h}_i^p; \boldsymbol{h}_i)$, as given in sequel

$$\mathcal{L}'(\boldsymbol{h}_i^p, \boldsymbol{h}_i) = -\log \frac{e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h}_i)/\tau}}{e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h}_i)/\tau} + \sum\limits_{\boldsymbol{h} \in \mathcal{B}^p} e^{\theta(\boldsymbol{h}_i^p, \boldsymbol{h})/\tau}}, \tag{3.10}$$

where the negative bank

$$\mathcal{B}^p = \{\boldsymbol{h}_j^p\}_{j \neq i} \cup \{\boldsymbol{h}_j\}_{j \neq i} \cup \{\widetilde{\boldsymbol{h}}_m^p\}_{m=1}^M \tag{3.11}$$

consists of all inter-view and intra-view negatives as well as synthesized hard negatives.

3.4 Model Training and Complexity Analysis.

The contrastive objective $\ell(\boldsymbol{h}_i; \boldsymbol{h}_i^p)$ for the aggregated node representation $\boldsymbol{h}_i$ can be defined similarly as Eq. (3.10). The final objective is an average of the losses from all contrastive pairs, formally given by

$$\mathcal{J} = \frac{1}{|\mathcal{V}|} \sum_{v_i \in \mathcal{V}} \left[\frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \frac{1}{2} \left(\mathcal{L}'(\boldsymbol{h}_i^p; \boldsymbol{h}_i) + \mathcal{L}'(\boldsymbol{h}_i; \boldsymbol{h}_i^p) \right) \right]. \tag{3.12}$$

We use stochastic gradient descent algorithms to update all model parameters. We summarize the training procedure in Appendix A in the supplementary material.

Most computational burden of the STENCIL framework lies in the contrastive objective, which involves computing $(|\mathcal{V}|^2|\mathcal{P}|)$ node embedding pairs. For structure-enhanced negative mining, the synthesized samples incur an additional computational cost of $O(M|\mathcal{V}||\mathcal{P}|)$, which is equivalent to increasing the memory size by $M \ll |\mathcal{V}|$. Since the construction of the candidate list of hard negatives only depends on metapath-induced views, it can be regarded as a preprocessing step and will not incur heavy computation.

3.5 Discussions.

The proposed multiview contrastive aggregation objective Eq. (3.6) conceptually relates to contrastive knowledge distillation [31], where several teacher

Table 1: Statistics and sources of the public datasets.

Dataset	Node	Relations	Metapaths
DBLP[1]	Paper (14,328) Author (4,057) Conference (20) Term (8,789)	P–A (19,645) P–C (14,328) P–T (88,420)	APA APCPA APTPA
ACM[2]	Paper (3,025) Author (5,835) Subject (56)	P–A (9,744) P–S (3,025)	PAP PSP
IMDb[3]	Movie (4,780) Actor (5,841) Director (2,269)	M–A (14,340) M–D (4,780)	MAM MDM

[1] http://ews.uiuc.edu/~jinggao3/doc/BGCM.zip

[2] https://github.com/Jhy1993/HAN/blob/master/data/acm/ACM.mat

[3] https://github.com/Jhy1993/HAN/blob/master/data/IMDb/movie_metadata.csv

models (the metapath-induced views) and one student model (the aggregation view) are employed. By forcing the embeddings between several teachers and a student to be the same, these aggregated embeddings adaptively collect information of all relations.

Moreover, the proposed structure-enhanced negative mining scheme generally resembles many studies in domains of metric learning [9, 41] and visual contrastive learning [1, 13, 23, 39]. Nevertheless, none of these methods can be directly applied to graph-structured data, as the hardness score defined simply by inner product of node representations is not sufficient to distinguish negative nodes in graphs and even results in amplifying false negatives.

4 Experiments

We empirically evaluate the effectiveness of our proposed STENCIL in this section. The purpose of empirical studies is to answer the following questions.

- **RQ1**. How does our proposed STENCIL outperform representative baseline models?
- **RQ2**. How does the proposed structure-enhanced negative mining scheme affect the model performance?

4.1 Datasets. For a comprehensive comparison, we use three widely-used heterogeneous datasets from various domains: DBLP, ACM, and IMDb, where DBLP and ACM are two academic networks, and IMDb is a movie network. The statistics of these three used datasets is summarized in Table 1. For details on datasets, please refer to Appendix B in the supplementary material.

4.2 Baselines. We compare our model against a comprehensive set of baselines, including both traditional and deep graph representation learning methods.

- **DeepWalk** [20] generates several sequences by random walk and training the embeddings using the skip-gram objective [17].
- **ESim** [26] captures node semantics from sampled metapath instances with a preset weight. In our experiments, we simply treat all metapaths equally.
- **metapath2vec** [4] performs metapath-based random walks and learns node representations using the skip-gram model similar to DeepWalk. Since metapath2vec only utilizes one metapath, we experiment with all available metapaths and report the best preformance.
- **HERec** [27] converts the heterogeneous graph into metapath-based graphs and utilizes the skip-gram model to embed them. Similarly, we test all metapaths and report the best performance.
- **GCN** [14] is a deep semi-supervised baseline for homogeneous graphs, which works by aggregating information from neighborhoods.
- **GAT** [34] is also a homogeneous graph model. It further leverages the self-attention mechanism to model anisotropic neighborhood information.
- **HAN** [36] is a semi-supervised baseline for heterogeneous graphs, which proposes node- and semantic-level attention for learning node representations. We also include the unsupervised version of HAN (denoted by **HAN-U**) trained with link prediction loss, for further comparison with our proposed contrastive objective.
- **DGI** [35] is a deep contrastive learning model for homogeneous graphs, which maximizes the agreement of node representations and a global summary vector.
- **GRACE** [46] is the state-of-the-art contrastive learning model for homogeneous graphs. It uses a node-level contrastive objective by generating two graph views and maximizing the agreement between them.
- **HeCo** [37] is the state-of-the-art heterogeneous contrastive learning model. HeCo constructs two views with metapaths and the network schema and leverages a collective contrastive scheme to align the two views.

Among these baselines, DeepWalk, DGI, GRACE, GCN, and GAT are designed for homogeneous graphs, and the others are for heterogeneous graphs. Following HAN [36], for DeepWalk, we simply discard node and edge types and treat the heterogeneous graph as a homogeneous graph; for DGI, GRACE, GCN, and GAT, we construct homogeneous graphs according to all metapaths and report the best performance.

4.3 Performance Comparison (RQ1)

4.3.1 Evaluation Protocols. For comprehensive evaluation, we follow HAN [36] and perform experiments on two

Table 2: Performance comparison on three datasets. We report node classification performance in terms of Macro-F1 (Ma-F1) and Micro-F1 (Mi-F1) and node clustering performance in terms of Normalized Mutual Information (NMI) and Adjusted Rand Index (ARI). Available training data is shown in the second column, where $\boldsymbol{A}$ denotes adjacency matrices according to metapaths, $\boldsymbol{X}$ denotes node features, and $\boldsymbol{Y}$ denotes ground-truth labels. The highest performance of unsupervised and supervised models is boldfaced and underlined, respectively.

Method	Training Data	Node Classification						Node Clustering					
		ACM		IMDb		DBLP		ACM		IMDb		DBLP	
		Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	NMI	ARI	NMI	ARI	NMI	ARI
DeepWalk	$\boldsymbol{A}$	76.92	77.25	46.38	40.72	79.37	77.43	41.61	35.10	1.45	2.15	76.53	81.35
ESim	$\boldsymbol{A}$	76.89	77.32	35.28	32.10	92.73	91.64	39.14	34.32	0.55	0.10	66.32	68.31
metapath2vec	$\boldsymbol{A}$	65.00	65.09	45.65	41.16	91.53	90.76	21.22	21.00	1.20	1.70	74.30	78.50
HERec	$\boldsymbol{A}$	66.03	66.17	45.81	41.65	92.69	91.78	40.70	37.13	1.20	1.65	**76.73**	78.50
HAN-U	$\boldsymbol{A}, \boldsymbol{X}$	82.63	81.89	43.98	40.87	90.47	89.65	39.84	32.98	3.92	4.10	74.17	79.98
DGI	$\boldsymbol{A}, \boldsymbol{X}$	89.15	89.09	48.86	45.38	91.30	90.69	58.13	57.18	8.31	11.25	60.62	60.42
GRACE	$\boldsymbol{A}, \boldsymbol{X}$	88.72	88.72	46.64	42.41	90.88	89.76	53.38	54.39	7.52	9.16	62.06	64.13
HeCo	$\boldsymbol{A}, \boldsymbol{X}$	88.15	88.25	51.69	50.75	91.56	91.02	59.53	57.59	10.11	11.74	70.99	76.67
STENCIL-PE	$\boldsymbol{A}, \boldsymbol{X}$	**90.76**	**90.72**	**58.98**	**54.48**	**92.81**	**92.33**	67.93	72.65	**15.09**	**17.23**	76.60	**81.58**
STENCIL-PPR	$\boldsymbol{A}, \boldsymbol{X}$	90.75	90.70	58.96	54.47	92.78	92.30	**68.10**	**73.15**	15.03	17.09	76.52	81.49
GCN	$\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{Y}$	86.77	86.81	49.78	45.73	91.71	90.79	51.40	53.01	5.45	4.40	75.01	80.49
GAT	$\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{Y}$	86.01	86.23	55.28	49.44	91.96	90.97	57.29	60.43	8.45	7.46	71.50	77.26
HAN	$\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{Y}$	89.22	89.40	54.17	49.78	92.05	91.17	61.56	64.39	10.31	9.51	79.12	84.76

tasks: node classification and node clustering. For node classification, we run a k-NN classifier with $k = 5$ on the learned node embeddings. We report performance in terms of Micro-F1 and Macro-F1 for evaluation of node classification. For dataset split, we randomly select 20% nodes in each dataset for training and the remaining 80% for test. Regarding node clustering, we run the k-Means algorithm on the learned node embeddings with k set to the number of ground-truth classes. We choose NMI and ARI of the obtained clusters with respect to ground-truth classes as the evaluation metrics for clustering. All experiments are repeated for 10 times and the averaged performance is reported.

4.3.2 Performance and Analysis. Experiment results are presented in Table 2. Overall, our proposed STENCIL achieves the best unsupervised performance on almost all datasets on both node classification and clustering tasks. It is worth mentioning that our STENCIL is competitive to and even better than several supervised counterparts.

The performance difference of two variants STENCIL-PE and STENCIL-PPR is negligible, which demonstrates that both Laplacian positional embedding and Personalized PageRank scores that calculate local structural similarities are suitable for structure-enhanced negative mining.

Compared with traditional approaches based on random walks and matrix decomposition, our proposed GNN-based STENCIL outperforms them by large margins. Particularly, STENCIL improves metapath2vec and HERec by over 25% on ACM, which demonstrates the superiority of GNN that can leverage rich node attributes to learn high quality node representations for heterogeneous graphs.

Compared to other deep unsupervised methods, our STENCIL obtains promising improvements as well. The performance of the unsupervised version HAN-U trained with a simple reconstruction loss is even inferior to HERec on IMDb and DBLP despite its utilization of node attributes. This indicates that the reconstruction loss is insufficient to fully exploit the structural and semantic information for node-centric tasks such as node classification and clustering. Compared to DGI, GRACE, and HeCo, state-of-the-art graph contrastive learning methods, STENCIL accomplishes excelled performance on all datasets and evaluation tasks, which validates the effectiveness of our proposed contrastive aggregation objective and structure-enhanced negative mining.

Furthermore, experiments show that STENCIL even outperforms its supervised baselines on ACM and IMDb datasets. It remarkably improves HAN by over 4% in terms of node classification Micro-F1 score on IMDb. The outstanding performance of STENCIL certifies the superiority of our proposed framework such that it can distill useful information from each metapath-induced view.

4.4 Close Inspections on Structure-Enhanced Negative Mining Module (RQ2)

4.4.1 Effectiveness of the Module. We modify the negative bank in our contrastive objective to study the impact of structure-enhanced negative mining component:

Table 3: Effectiveness of the structure-enhanced negative mining module.

Method	Node Classification						Node Clustering					
	ACM		IMDb		DBLP		ACM		IMDb		DBLP	
	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	NMI	ARI	NMI	ARI	NMI	ARI
STENCIL–	88.62	88.43	57.94	52.97	92.42	91.85	58.08	61.80	14.15	15.98	76.23	81.43
STENCIL-Sem	90.24	90.18	58.95	52.38	92.73	92.21	51.63	48.85	15.17	17.25	76.22	81.15
STENCIL-PE	**91.40**	**91.45**	**58.96**	**53.73**	**92.77**	**92.28**	**66.57**	**72.30**	**15.36**	**17.30**	**76.59**	**81.56**

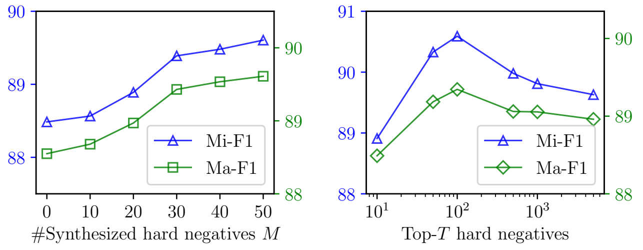
- STENCIL– denotes the model with synthesized hard samples $\{\widetilde{\boldsymbol{h}}_m^p\}_{m=1}^M$ removed, where the negative bank $\mathcal{B}^p = \{\boldsymbol{h}_j^p\}_{j\neq i} \cup \{\boldsymbol{h}_j\}_{j\neq i}$ consists of only inter- and intra-view negatives.
- STENCIL-Sem discovers and synthesizes semantic negative samples using inner product of node embeddings, which is similar to earlier visual contrastive learning work MoChi [13].

The results are presented in Table 3. It is observed that STENCIL improves all two model variants consistently on three datasets for both node classification and clustering tasks. Especially for node clustering task on ACM, the gain reaches up to 15%. This verifies the effectiveness of our synthesizing hard negative sample strategy using structure embeddings. Secondly, we see that the performance of STENCIL-Sem occasionally improves the base model, which demonstrates the importance of hard negative mining in effective contrastive learning. However, its performance is still inferior to that of our proposed model. The outstanding performance of STENCIL compared to the model variant STENCIL-Sem further justifies the superiority of our proposed structure-enhanced negative mining, which exploits the abundant structural information of HGs.

4.4.2 The Impact of Two Key Parameters in the Module.

We study how the two key parameters in the negative mining module affect the performance of STENCIL: the number of synthesized hard negatives M and the threshold T in selecting top-T candidate hard negatives. We perform node classification on the ACM dataset under different parameter settings by only varying one specific parameter and keeping all other parameters the same.

As is shown in Figure 4a, the performance of STENCIL improves as the number of synthesized negatives M increases, which indicates that the learning of STENCIL benefits from more synthesized hard negatives. For the parameter T, as presented in Figure 4b, the model performance first rises with a larger T, but soon the performance levels off and decreases as T increases further. We suspect that this is because a larger T will result in the selection of less hard negatives, reducing the benefits brought by our proposed structure-enhanced negative mining strategy.



(a) Synthesized hard negatives (b) Candidate hard negative samples

Figure 4: Node classification performance with varied numbers of synthesized hard negatives and candidate hard negative samples T on the ACM dataset.

5 Related Work

This section reviews previous related work on heterogeneous graph embedding methods. Following that, we discuss recent work on graph contrastive learning.

5.1 Heterogeneous Graph Embedding.

The purpose of Heterogeneous Graph Embedding (HGE) is to project nodes in a heterogeneous graph into a low-dimensional embedding space that preserves structural and semantic information.

Proximity-preserving methods. Inspired by network embedding methods for homogeneous graphs, traditional HGE methods roughly fall into two lines: random-walk-based approaches and methods based on preserving first-/second-order proximity. On the one hand, originated from random-walk-based methods for homogeneous graphs [8, 20], metapath2vec [4] models node context via metapath-based random walks and learns node embeddings using the skip-gram model [17]. Similarly, HERec [27] transforms a heterogeneous graph into a homogeneous one through metapath-based neighborhood and learns representations using DeepWalk-like strategies. HIN2Vec [6] further proposes a multitask learning objective to learn representations for nodes and metapaths simultaneously. On the other hand, the pioneering proximity-preserving method PTE [30] extends LINE [30] to heterogeneous text graphs. HEER [28] further improves PTE by considering type closeness via edge representations. These aforementioned traditional approaches could be regarded as shallow embedding and thus have difficulty in leveraging rich node attributes, due to the fact that they are essentially factorizing a preset proximity matrix [21].

Deep learning approaches. There has been many attempts adopting GNNs into HGs. R-GCN [24] introduces multiple graph convolutional layers, each corresponding to one edge type. GTN [43] firstly generates all possible connections via graph transformer layers and performs graph convolution on the new graph afterwards. Following GAT [34], HAN [36] introduces self-attention mechanisms [33] to aggregate features from metapath-based neighborhoods and weigh different metapaths. Similarly, HetGNN [44] adopts node-type-based neighborhood aggregation, where the neighborhood is sampled using random walk with restart. Moreover, MAGNN [7] further proposes to aggregate intermediate node features along each metapath. When performing neighborhood aggregation, HGT [11] implicitly learns metapaths by modeling heterogeneous attention over each edge.

5.2 Graph Contrastive Learning. Recently, considerable attention has grown up around the theme of Graph Contrastive Learning (GCL), which marries the power of GNN and unsupervised learning. We refer readers to [38] for a comprehensive survey.

The very first work DGI [35] proposes to maximizes mutual information (ML) between node embeddings and a global summary embedding. To be specific, DGI constructs negative graphs by random shuffling node attributes. Then, it requires an injective readout function to produce a graph-level embedding. Mirroring DGI, HDGI [22] adopts CL into heterogeneous graphs. DMGI [19] proposes to align the original network and a corrupted network on each view induced by metapaths and designs a consensus regularization term to aggregate different metapaths. However, the injective property is hard to fulfill in practice and thus these methods may cause information loss due to non-injectivity. Follow-up work GRACE [46] eschews the need of an injective readout function and propose a node-level contrastive framework. Following their work, GCA [47] further proposes stronger adaptive augmentation schemes.

For HG data, prior work proposes various strategies for the contrastive objective. HeCo [37] constructs two views based on metapaths and the network schema and proposes a co-contrastive objective to learn high-level semantic information. Nevertheless, these methods fail to explicitly leverage structural information for GCL, leading to suboptimal performance. We argue that inner product of node embeddings is inefficient to encode similarity between nodes. In our work, we propose to define similarity of examples via structural embeddings, which yields true and hard negative samples in the context of HGs.

6 Conclusion

This paper has developed a novel heterogeneous graph contrastive learning framework. To alleviate the label scarcity problem, we leverage contrastive learning techniques that enables self-supervised training for HGs. Specifically, we propose a novel multiview contrastive aggregation objective that encodes information adaptively from each semantic view. Furthermore, we propose a novel hard negative mining scheme to improve the embedding quality, considering the complex structure of HGs and smoothing nature of heterogeneous GNNs. The proposed structure-aware negative mining scheme discovers and reweights structurally hard negatives so that they contribute more to contrastive learning. Extensive experiments show that our proposed method not only consistently outperforms representative unsupervised baseline methods, but also achieves on par performance with supervised counterparts and even surpasses several of them.

Acknowledgements

This work is supported by National Natural Science Foundation of China (61772528 and U19B2038).

References

[1] T. T. Cai, J. Frankle, D. J. Schwab, and A. S. Morcos, *Are All Negatives Created Equal in Contrastive Instance Discrimination?*, arXiv.org, (2020), https://arxiv.org/abs/2010.06682v2.

[2] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, *A Simple Framework for Contrastive Learning of Visual Representations*, in ICML, 2020, pp. 1597–1607.

[3] H. Cui, Z. Lu, P. Li, and C. Yang, *On Positional and Structural Node Features for Graph Neural Networks on Non-attributed Graphs*, arXiv.org, (2021), https://arxiv.org/abs/2107.01495v1.

[4] Y. Dong, N. V. Chawla, and A. Swami, *metapath2vec: Scalable Representation Learning for Heterogeneous Networks*, in KDD, 2017, pp. 135–144.

[5] V. P. Dwivedi, C. K. Joshi, T. Laurent, Y. Bengio, and X. Bresson, *Benchmarking Graph Neural Networks*, arXiv.org, (2020), https://arxiv.org/abs/2003.00982v3.

[6] T.-Y. Fu, W.-C. Lee, and Z. Lei, *HIN2Vec: Explore Metapaths in Heterogeneous Information Networks for Representation Learning*, in CIKM, 2017, pp. 1797–1806.

[7] X. Fu, J. Zhang, Z. Meng, and I. King, *MAGNN: Metapath Aggregated Graph Neural Network for Heterogeneous Graph Embedding*, in WWW, 2020, pp. 2331–2341.

[8] A. Grover and J. Leskovec, *node2vec: Scalable Feature Learning for Networks*, in KDD, 2016, pp. 855–864.

[9] B. Harwood, B. G. V. Kumar, G. Carneiro, I. D. Reid, and T. Drummond, *Smart Mining for Deep Metric Learning*, in ICCV, 2017, pp. 2840–2848.

[10] O. J. Hénaff, A. Srinivas, J. De Fauw, A. Razavi, C. Doersch, S. M. A. Eslami, and A. van den Oord,

Data-Efficient Image Recognition with Contrastive Predictive Coding, in ICML, 2020, pp. 4182–4192.

[11] Z. Hu, Y. Dong, K. Wang, and Y. Sun, *Heterogeneous Graph Transformer*, in WWW, 2020, pp. 2704–2710.

[12] G. Jeh and J. Widom, *Scaling Personalized Web Search*, in WWW, 2003.

[13] Y. Kalantidis, M. B. Sariyildiz, N. Pion, P. Weinzaepfel, and D. Larlus, *Hard Negative Mixing for Contrastive Learning*, in NeurIPS, 2020.

[14] T. N. Kipf and M. Welling, *Semi-Supervised Classification with Graph Convolutional Networks*, in ICLR, 2017.

[15] P. Li, Y. Wang, H. Wang, and J. Leskovec, *Distance Encoding: Design Provably More Powerful Neural Networks for Graph Representation Learning*, in NeurIPS, 2020.

[16] G. Mialon, D. Chen, M. Selosse, and J. Mairal, *GraphiT: Encoding Graph Structure in Transformers*, arXiv.org, (2021), https://arxiv.org/abs/2106.05667v1.

[17] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, *Efficient Estimation of Word Representations in Vector Space*, in ICLR, 2013.

[18] L. Page, S. Brin, R. Motwani, and T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web*, tech. report, Nov. 1999.

[19] C. Park, D. Kim, J. Han, and H. Yu, *Unsupervised Attributed Multiplex Network Embedding*, in AAAI, 2020, pp. 5371–5378.

[20] B. Perozzi, R. Al-Rfou, and S. Skiena, *DeepWalk: Online Learning of Social Representations*, in KDD, 2014, pp. 701–710.

[21] J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, *Network Embedding as Matrix Factorization: Unifying DeepWalk, LINE, PTE, and node2vec*, in WSDM, 2018, pp. 459–467.

[22] Y. Ren, B. Liu, C. Huang, P. Dai, L. Bo, and J. Zhang, *Heterogeneous Deep Graph Infomax*, arXiv.org, (2019), https://arxiv.org/abs/1911.08538v5.

[23] J. Robinson, C.-Y. Chuang, S. Sra, and S. Jegelka, *Contrastive Learning with Hard Negative Samples*, arXiv.org, (2020), https://arxiv.org/abs/2010.04592v1.

[24] M. S. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, *Modeling Relational Data with Graph Convolutional Networks*, in ESWC, 2018, pp. 593–607.

[25] F. Schroff, D. Kalenichenko, and J. Philbin, *FaceNet: A Unified Embedding for Face Recognition and Clustering*, in CVPR, 2015, pp. 815–823.

[26] J. Shang, M. Qu, J. Liu, L. M. Kaplan, J. Han, and J. Peng, *Meta-Path Guided Embedding for Similarity Search in Large-Scale Heterogeneous Information Networks*, arXiv.org, (2016), https://arxiv.org/abs/1610.09769v1.

[27] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu, *Heterogeneous Information Network Embedding for Recommendation*, TKDE, 31 (2019), pp. 357–370.

[28] Y. Shi, Q. Zhu, F. Guo, C. Zhang, and J. Han, *Easing Embedding Learning by Comprehensive Transcription of Heterogeneous Information Networks*, in KDD, 2018, pp. 2190–2199.

[29] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, *Co-author Relationship Prediction in Heterogeneous Bibliographic Networks*, in ASONAM, 2011, pp. 121–128.

[30] J. Tang, M. Qu, and Q. Mei, *PTE: Predictive Text Embedding through Large-scale Heterogeneous Text Networks*, in KDD, 2015, pp. 1165–1174.

[31] Y. Tian, D. Krishnan, and P. Isola, *Contrastive Representation Distillation*, in ICLR, 2020.

[32] A. van den Oord, Y. Li, and O. Vinyals, *Representation Learning with Contrastive Predictive Coding*, arXiv.org, (2018), https://arxiv.org/abs/1807.03748v2.

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, U. Kaiser, and I. Polosukhin, *Attention is All You Need*, in NIPS, 2017, pp. 5998–6008.

[34] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, *Graph Attention Networks*, in ICLR, 2018.

[35] P. Veličković, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, *Deep Graph Infomax*, in ICLR, 2019.

[36] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, *Heterogeneous Graph Attention Network*, in WWW, 2019, pp. 2022–2032.

[37] X. Wang, N. Liu, H. Han, and C. Shi, *Self-supervised Heterogeneous Graph Neural Network with Co-contrastive Learning*, in KDD, 2021, pp. 1726–1736.

[38] L. Wu, H. Lin, Z. Gao, C. Tan, and S. Z. Li, *Self-supervised on Graphs: Contrastive, Generative, or Predictive*, arXiv.org, (2021), https://arxiv.org/abs/2105.07342v1.

[39] M. Wu, M. Mosse, C. Zhuang, D. Yamins, and N. Goodman, *Conditional Negative Sampling for Contrastive Learning of Visual Representations*, in ICLR, 2021.

[40] J. Xia, L. Wu, J. Chen, G. Wang, and S. Z. Li, *Debiased Graph Contrastive Learning*, arXiv.org, (2021), https://arxiv.org/abs/2110.02027v1.

[41] H. Xuan, A. Stylianou, X. Liu, and R. Pless, *Hard Negative Examples are Hard, but Useful*, in ECCV, 2020, pp. 126–142.

[42] J. You, R. Ying, and J. Leskovec, *Position-aware Graph Neural Networks*, in ICML, June 2019, pp. 7134–7143.

[43] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, *Graph Transformer Networks*, in NeurIPS, 2019, pp. 11960–11970.

[44] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, *Heterogeneous Graph Neural Network*, in KDD, 2019, pp. 793–803.

[45] Y. Zhu, Y. Xu, Q. Liu, and S. Wu, *An Empirical Study of Graph Contrastive Learning*, in NeurIPS Datasets and Benchmarks, 2021.

[46] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, *Deep Graph Contrastive Representation Learning*, in GRL+@ICML, June 2020.

[47] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, and L. Wang, *Graph Contrastive Learning with Adaptive Augmentation*, in WWW, 2021, pp. 2069–2080.