GLOBAL AND NODAL MUTUAL INFORMATION MAXIMIZATION IN HETEROGENEOUS GRAPHS

Costas Mavromatis and George Karypis

Department of Computer Science, University of Minnesota, USA

ABSTRACT

Many real-world graphs involve different types of nodes and edges, being heterogeneous by nature. Heterogeneous graph representation learning embeds their rich structure and semantics into a low-dimensional space to facilitate graph related tasks. In this work, we propose a self-supervised method that learns representations by relying on mutual information maximization among different graph structures (metapaths). Our method, termed HeMI, promotes node-level and global-level shared semantics among nodes with contrastive learning, as well as it leverages interactions among metapaths. Experiments on node classification, node clustering, and link prediction show that HeMI outperforms existing approaches.

***Index Terms*—** Heterogeneous Graphs, Graph Representation Learning, Self-Supervised Learning

1. INTRODUCTION

Heterogeneous graphs (HGs) model compositions of different types of nodes and edges, which emerge in various real-world applications. An illustrative example of a HG is shown in Fig. 1 (left) and other examples include bibliographic networks, social networks, and recommendation systems. In this work, we focus on HG representation learning (HGRL), which encodes this high-dimensional, non-Euclidean, and heterogeneous information to a low-dimensional Euclidean embedding space. The learned representations facilitate various tasks, such as node classification, node clustering, and link prediction. Moreover, *self-supervised* HGRL eliminates the need of obtaining costly training labels, and allows to reuse the learned representations for different tasks as well as to fine-tune on specific tasks with few labeled data.

When dealing with homogeneous graphs, various GRL methods have been proposed. Approaches, such as DeepWalk [1], node2vec [2] and Mazi [3], optimize the representations so that nodes preserve their structural properties. Since these methods are not inherently designed to handle node attributes, graph neural networks (GNNs) [4, 5, 6] have been proposed, which are deep learning methods designed for handling both the graph structure and the node attributes. GNNs mainly estimate node representations through a recursive neighborhood aggregation scheme [7]. As GNNs handle

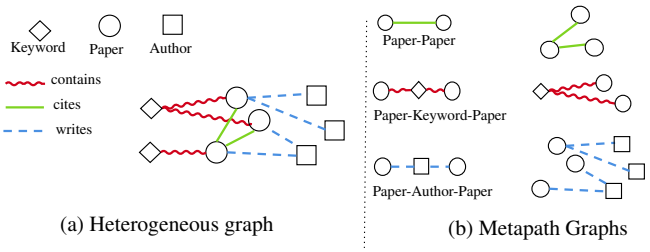

Fig. 1. An example of a HG (citation network) (left) and its metapath induced graphs (right).

the structural properties of the graph, recent self-supervised GRL methods rely on preserving higher order similarities of nodes. DGI [8] encodes information that is shared among all nodes of the graph, while GIC [9] encodes shared information within clusters of nodes.

A natural approach for HGRL is to extend homogeneous-designed methods to handle the heterogeneity of the data. This is typically done by modelling compositions of HG relations as metapaths [10]. Each metapath captures different HG semantics and can be treated as a specific homogeneous view of the HG. An example is shown in Fig. 1 (right). Consequently, methods such as metapath2vec [11] and HERec [12] perform random walks over metapaths and Heterogeneous GNNs (HGNNs), such as RGCN [13], HAN [14], and MAGNN [15], encode the structural properties of the metapaths. Applications of HGRL range from citations networks [14] and recommendation systems [12] to biological networks [16] and drug interactions [17].

Recent self-supervised HGRL methods combine HGNNs and higher order self-supervision signals to learn node representations. HDGI [18] acts in similar manner with DGI to promote globally shared semantics at the final node representations. DMGI [19] extends DGI by employing an additional consensus regularization among metapaths. HDMI [20] enhances DGI with the input node features. Recent methods, such as HeCo [21] and STINCEL [22], follow the multi-view paradigm [23] to promote node-level consensus among different metapaths. However, combining both global and node-level shared semantics among metapaths has not been well-explored.

Contributions. In this work, we propose a unified self-supervised HGRL objective that captures both global and node-level shared semantics across metapaths. Following the mutual information maximization framework (MIM) [24, 8], we promote global and node-level consensus among metapaths with contrastive learning [24, 8]. Global MI preserves metapath-specific properties, while nodal MI preserves node-level properties.

Metapaths are usually treated independently, which may lose information about the way they interact. Thus, we propose a simple, yet effective, technique that couples metapath semantics and promotes additional consensus among them. Experiments on three benchmarks showcase our method improves over existing approaches by up to 2.49% points for node clustering and up to 13.7% points for link prediction.

2. PRELIMINARY

Heterogeneous graph [25]. A heterogeneous graph (HG), denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, consists of a node set $\mathcal{V} = \{\mathcal{V}\}_{t=1}^{T}$ and an edge set $\mathcal{E} = \{\mathcal{E}\}_{r=1}^{R}$. It is used to represent a network with T types of nodes and R types of edges. Fig. 1 shows a heterogeneous graph with 3 node types and 3 edge types.

Metapath [10]. A metapath P_j is defined as a path in the form $V_1 \xrightarrow{E_1} V_2 \xrightarrow{E_2} \cdots \xrightarrow{E_{l-1}} V_l$ (abbreviated as $V_1V_2\cdots V_l$), which describes a composite relation $E = E_1 \circ \cdots \circ E_{l-1}$ between node types V_1 and V_l. An example metapath of Fig. 1 is Paper $\xrightarrow{\text{written}}$ Author $\xrightarrow{\text{writes}}$ Paper (PAP).

The metapath-based graph $\mathcal{G}^{P_j}$ is a graph constructed from all node pairs $v \in \mathcal{V}_1$ and $u \in \mathcal{V}_l$ that connect via metapath P_j. Metapath-based graphs exploit different aspects of structural information in HGs. We represent each $\mathcal{G}^{P_j}$ with an adjacency matrix $\boldsymbol{A}^j$, with $\boldsymbol{A}^j_{uv} = 1$ if edge $(u, v) \in \mathcal{G}^{P_j}$ and 0 otherwise.

HGRL. We focus on learning representations of a target node type, denoted as $\mathcal{V}_\tau$, where the total number of target nodes is N. Given a HG $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with target node attribute matrix $\mathbf{X} \in \mathbb{R}^{N\times d}$, self-supervised HGRL is the task of learning the d-dimensional node representations $\mathbf{z}_v \in \mathbb{R}^d$ for all $v \in \mathcal{V}_\tau$. It is desired that the representations are able to capture rich structural and semantic information involved in $\mathcal{G}$ without using training labels. Following existing literature [11, 18, 19], we use $\mathcal{G}$ to obtain metapath-based graphs $\{\mathcal{G}^{P_j}\}_{j=1}^{M}$ that have the target nodes as starting and ending nodes.

Mutual Information. Mutual information (MI) is based on Shannon entropy and measures the mutual dependence between two random variables X and Y, denoted as $I(X;Y)$. Since MI is in general intractable, it is often estimated by parametrized functions, such as a deep neural networks.

3. METHODOLOGY

As metapaths contain useful information (they could be provided by experts or data engineers), we learn node representations that preserve this information. We rely on the mutual information maximization framework (MIM) and promote information that is shared across different metapaths for the same node (Section 3.3). As some nodes may contain noisy information, we employ a global MIM across nodes of the same metapath to promote globally shared properties (Section 3.4). Moreover, to better preserve interactions among metapaths, we employ a simple, yet effective, technique that couples metapath representations (Section 3.2). We call our method HeMI as it employs a Heterogenous MI Maximization objective.

3.1. Metapath Representations

To capture the unique structural properties and attributes of each metapath graph $\mathcal{G}^{P_j}$, we employ a GNN encoder and learn metapath-specific node representations. We use an 1-layer GCN [4] encoder and compute the node representation matrix

$$\mathbf{Z}^j = \sigma(\bar{\mathbf{A}}^j\mathbf{X}\mathbf{W}^j), \tag{1}$$

where $\bar{\mathbf{A}}^j$ is the normalized adjacency matrix of metapath P^j as defined in [18], $\mathbf{W}^j$ are learnable parameters, and $\sigma(\cdot)$ is a nonlinear activation such as PReLU.

To combine information from different metapaths, we aggregate the representations $\{\mathbf{z}_v^j\}_{j=1}^{M}$, where $\mathbf{z}_v^j$ is the representation of node v for metapath P^j. We compute the fused representation $\mathbf{z}_v$ as

$$\mathbf{z}_v = \sum_j \beta^j \mathbf{z}_v^j, \tag{2}$$

where the coefficients β^j are computed by an attention mechanism [14, 15] that learns the importance of each meta-path. The attention weights β^j are obtained by applying

$$\mathbf{e}^j = \frac{1}{N}\sum_v \mathbf{q}^T\mathbf{W}_q\mathbf{z}_v^j, \text{ and } \beta^j = \frac{\exp(\mathbf{e}^j)}{\sum_j \exp(\mathbf{e}^j)}, \tag{3}$$

where $\mathbf{W}_q$ are learnable parameters, and $\mathbf{q}$ is a parametrized attention vector.

3.2. Coupled Metapath Representations (CMR)

Note that Eq.(1) treats each metapath independently during the GCN updates. However, this might lose important information on how different metapaths interact. For example, metapaths Paper-Conference-Paper and Paper-Subject-Conference-Paper have very similar semantics. Thus, we use

a shared GCN encoder among all metapaths to compute

$$\mathbf{H}^j = \sigma(\bar{\mathbf{A}}^j \mathbf{X} \mathbf{W}_\mathrm{s}), \tag{4}$$

$$\mathbf{h}_v = \sum_j \gamma^j \mathbf{h}_v^j, \tag{5}$$

where γ^j coefficients are computed similarly to Eq.(3). Because $\mathbf{h}_v^j$ are computed by the same GCN (shared parameters $\boldsymbol{W}_\mathrm{s}$), all metapaths are encoded in a shared representation space.

3.3. Nodal MI Across Metapaths

A straightforward way to ensure that we encode information from all metapaths is to maximize the MI between each $\mathbf{z}_v^j$ with $\mathbf{z}_v$ as

$$\max \sum_j I(\mathbf{z}_v^{P_j}; \mathbf{z}_v). \tag{6}$$

In order to estimate the MI we use contrastive learning [24, 8] to discriminate between positive and negative examples. Positive examples are pairings of $(\mathbf{z}_v^j, \mathbf{z}_v)$, and negatives are pairings of $(\mathbf{z}_v^j, \tilde{\mathbf{z}}_v)$. We obtain $\tilde{\mathbf{z}}_v$ as the output of our model with fake features $\tilde{\boldsymbol{X}}$ as input. We obtain $\tilde{\boldsymbol{X}}$ by replacing the attributes of each node with the attributes of a randomly selected node. To optimize model's parameters, we employ the contrastive loss

$$\mathcal{L}_n = \sum_j \mathbb{E}[\log D(\boldsymbol{z}_v^j, \boldsymbol{z}_v)] + \mathbb{E}[\log\big(1 - D(\boldsymbol{z}_v^j, \tilde{\boldsymbol{z}}_v)\big)] \tag{7}$$

which acts as a binary cross-entropy loss, where $D_n(\cdot)$ is the discriminator. We implement the discriminator as a bilinear function, followed by a sigmoid $\sigma(\cdot)$ that transforms scores to probabilities, i.e., $D(\boldsymbol{z}_v^j, \boldsymbol{z}_v) = \sigma(\mathbf{z}_v^j \boldsymbol{W}_D \mathbf{z}_v)$. Similarly, we employ

$$\mathcal{L}_n^c = \sum_j \mathbb{E}[\log D(\boldsymbol{h}_v^j, \boldsymbol{h}_v)] + \mathbb{E}[\log\big(1 - D(\boldsymbol{h}_v^j, \tilde{\boldsymbol{h}}_v)\big)], \tag{8}$$

to maximize nodal MI across the coupled representations $\boldsymbol{h}_v^j$ of Section 3.2.

3.4. Global MI Across Nodes

MI of Eq.(6) can be maximized by simply preserving noisy information of $\boldsymbol{z}_v^j$. To encode information that is globally present, i.e., semantically shared properties, we first compute global metapath representations $\boldsymbol{s}^j$ (decoupled) and $\boldsymbol{g}^j$ (coupled) by averaging the metapath-specific node representations as

$$\boldsymbol{s}^j = \sigma\big(\frac{1}{N}\sum_v \boldsymbol{z}_v^j\big) \text{ and } \boldsymbol{g}^j = \sigma\big(\frac{1}{N}\sum_v \boldsymbol{h}_v^j\big). \tag{9}$$

Then, we employ the maximization objective

$$\max \sum_j I(\boldsymbol{s}^j; \mathbf{z}_v) + \sum_j I(\boldsymbol{g}^j; \mathbf{h}_v) \tag{10}$$

Table 1. Dataset statistics.

Dataset	Nodes	Edges	Meta-paths	Features
ACM	Paper: 3,025 Author: 5,835 Subject: 56	P-A: 9,744 P-S: 3,025	PAP PSP	1,870
DBLP	Author: 4,057 Paper: 14,328 Conference: 20 Term: 8,789	A-P: 19,645 P-C: 14,328 P-T: 88,420 P-T: 85,810	APA APCPA APTPA	334
IMDB	Movie: 4,275 Director: 2,082 Actor: 5,431 Keyword: 7,313	M-A: 12,838 M-D: 4,280 M-K: 20,529	MAM MDM MKM	5,927

Table 2. Node classification (% MicroF1) and node clustering (% NMI) performance for different methods.

	Node Classification			Node Clustering		
Method	ACM	DBLP	IMDB	ACM	DBLP	IMDB
GCN	92.60	92.04	59.53	71.05	73.45	7.46
HAN	92.81	93.27	61.23	73.25	77.49	10.79
metap2vec / DeepWalk	81.78	88.09	56.36	41.15	34.30	5.97
SSMGRL	88.61	91.71	53.99	45.24	73.27	2.2
STENCIL / HeCo	90.76	92.81	-	68.10	**76.60**	-
MNI-DGI / DMGI	93.19	92.79	60.33	68.01	75.06	2.18
GIC / DGI	92.94	93.22	62.40	69.37	69.08	8.18
HGIC / HDGI	93.41	92.64	59.59	70.51	69.68	1.87
HeMI	**93.65**	**93.37**	**63.62**	71.09	75.70	**10.57**
Global MI	**93.65**	93.15	63.18	70.35	73.32	8.59
Nodal MI	93.45	**93.37**	63.30	71.09	75.70	9.27
HeMI w/o CMR	**93.66**	92.96	59.57	**71.76**	70.05	1.64
Global MI	93.35	92.90	58.97	71.63	69.84	0.63
Nodal MI	**93.66**	92.09	59.57	**71.76**	68.88	0.80

to promote globally shared semantics among nodes of the same metapath. If a node has irrelevant information, this information will not maximize the global MI and thus will not be preferred. Similar to Eq.(7) and Eq.(8), we maximize Eq.(10) by employing

$$\mathcal{L}_g = \sum_j \mathbb{E}[\log T(\boldsymbol{s}^j, \boldsymbol{z}_v)] + \mathbb{E}[\log(1 - T(\boldsymbol{s}^j, \tilde{\boldsymbol{z}}_v))], \tag{11}$$

$$\mathcal{L}_g^c = \sum_j \mathbb{E}[\log T(\boldsymbol{g}^j, \boldsymbol{h}_v)] + \mathbb{E}[\log(1 - T(\boldsymbol{g}^j, \tilde{\boldsymbol{h}}_v))], \tag{12}$$

where $T(\cdot)$ is again a discriminator with parameters $\boldsymbol{W}_T$.

We combine nodal MI objectives $\mathcal{L}_n$ and $\mathcal{L}_n^{(c)}$ with global MI objectives $\mathcal{L}_g$ and $\mathcal{L}_g^{(c)}$ to a unified objective $\mathcal{L}$, which is given by

$$\mathcal{L} = \lambda\big(\mathcal{L}_n + \mathcal{L}_n^{(c)}\big) + (1-\lambda)\big(\mathcal{L}_g + \mathcal{L}_g^{(c)}\big). \tag{13}$$

Hyper-parameter $\lambda \in [0, 1]$ controls the relative importance of the nodal and global MI. The final node representation of node v is the concatenation of $\boldsymbol{z}_v$ and $\boldsymbol{h}_v$.

4. EXPERIMENTS

4.1. Datasets and Baselines

To demonstrate the effectiveness of HeMI over existing appraches, we conduct experiments on three public benchmark

Table 3. Link Prediction results (% AUC/AP) for different objectives.

Objective	ACM AUC / AP	DBLP AUC / AP	IMDB AUC / AP
w/o CMR			
Link Prediction	66.3 / 65.0	71.5 / 72.6	69.7 / 72.7
HeMI + LP	62.3 / 66.2	72.7 / 74.5	70.6 / 73.0
w/ CMR			
Link Prediction	68.9 / 60.7	77.3 / 79.3	73.5 / 67.6
HeMI + LP	**74.1 / 67.2**	**84.2 / 85.9**	**81.9 / 81.3**

datasets, including ACM, DBLP, and IMDB. ACM and DBLP are categorized into paper areas, while IMDB into movie genres. We follow the setup of [18] and the dataset statistics are shown in Table 1.

We evaluate the performance of our model against well established GRL methods (DeepWalk [1], GIC [9], DGI [8]) as well as HGRL methods (metap2vec [11], SSMGRL [26], DMGI [19], MNI-DGI [27], HGIC [9], HDGI [18]). Due to space limitations, we group similar methods together, e.g., 'GIC / DGI', and report their best result. We also report results for GCN [4] and HAN [14] that are semi-supervised GNNs that use labels. For HeMI, 'Nodal MI' means that we have $\lambda = 1$ in Eq.(13), while 'Global MI' means $\lambda = 0$. 'w/o CMR' means that we omit terms $\mathcal{L}_n^{(c)}$ and $\mathcal{L}_g^{(c)}$ that use the coupled representations of Section 3.2. We optimize model's parameters with Adam and early stopping, and set $d = 256$. Further details can be found in our code at: `https://github.com/cmavro/HeMI`.

4.2. Node Classification and Clustering

For node classification, we use the generated node representations to train a linear classifier with 20% training nodes (10% validation and 70% test nodes) and report Micro-F1 as the metric (averaged over 10 runs). For node clustering, we apply K-means to generate clusters and report normalized mutual information (NMI) as the metric.

As Table 2 shows, our method outperforms other baselines for both node classification and node clustering. For DBLP, node-level MIM approaches perform the best (HeMI, and STENCIL), while for IMDB metapath global semantics are more important (HeMI, GIC, and DGI). For DBLP and IMDB, some metapaths are closely related, e.g., APA with APTPA in DBLP and MAM with MDM in IMDB. Thus, coupling metapaths improves HeMI's node clustering by 5.65% and 8.93% points for DBLP and IMDB, respectively. In most cases, HeMI's Nodal MI is more important than Global MI, but the performance is improved when complemented with Global MIM.

4.3. Link Prediction

In link prediction, node representations are used to predict missing links. We sample 25% (20% for testing and 5% for validation) positive and negative edges of each metapath and use the learned metapath node representations to predict them. As a baseline, we use the reconstruction objective of GAE [28] that learns representations that preserve the existing edges with link prediction (LP). We report the area under curve (AUC) and average precision (AP), averaged over all metapaths for 5 runs.

Table 3 shows that complementing link prediction with HeMI objective (HeMI + LP) improves the overall performance when predicting missing links for each metapath. For DBLP and IMDB, HeMI improves over LP by 6.9% to 13.7% points as it combines information from different metapaths that is beneficial for the missing links (LP treats the metapaths independently). Moreover, CMR capture additional interactions among metapaths and improve both HeMI and LP by up to 11.3% and 6.7% points, respectively. For ACM, fusing metapath information is not that crucial, thus methods behave similarly (as also observed in Table 2).

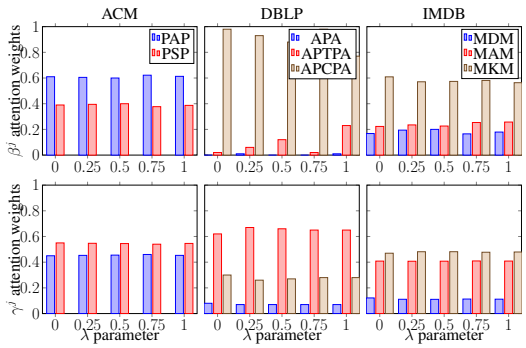

Fig. 2. Learned attention weights of Eq.(2) and Eq.(5) (y-axes) for each meta-path with respect to λ values (x-axis).

4.4. Ablation Study

To show how Nodal, Global MIM, and CMR interact, Fig. 2 visualizes the learned metapath attention weights with respect to different HeMI's variants. For uncoupled metapath representations (top figures), HeMI attends more to specific metapaths (PAP for ACM, APCPA for DBLP, and MKM for IMDB). On the other, for CMR (bottom figures), attention weights are more uniformly distributed. We observe similar trends for different λ values.

5. CONCLUSION

We have presented HeMI, a HGRL approach that relies on MI maximization to learn useful node representations. Experiments showed that HeMI's Nodal and Global MI benefit HG tasks such as node classification and node clustering. Moreover, HeMI's CMR improve performance when metapath interactions are important, as in link prediction.

6. REFERENCES

[1] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena, "Deepwalk: Online learning of social representations," in *KDD*, 2014.

[2] Aditya Grover and Jure Leskovec, "node2vec: Scalable feature learning for networks," in *KDD*, 2016.

[3] Ancy Sarah Tom, Nesreen K Ahmed, and George Karypis, "Joint learning of hierarchical community structure and node representations: An unsupervised approach," *ECML-PKDD*, 2022.

[4] Thomas N Kipf and Max Welling, "Semi-supervised classification with graph convolutional networks," *ICLR*, 2017.

[5] Will Hamilton, Zhitao Ying, and Jure Leskovec, "Inductive representation learning on large graphs," in *NeurIPS*, 2017.

[6] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio, "Graph Attention Networks," *ICLR*, 2018.

[7] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl, "Neural message passing for quantum chemistry," in *ICML*, 2017.

[8] Petar Veličković, William Fedus, William L. Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm, "Deep Graph Infomax," in *ICLR*, 2019.

[9] Costas Mavromatis and George Karypis, "Graph infoclust: Maximizing coarse-grain mutual information in graphs," *PAKDD*, 2021.

[10] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *VLDB*, 2011.

[11] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *KDD*, 2017.

[12] Chuan Shi, Binbin Hu, Wayne Xin Zhao, and S Yu Philip, "Heterogeneous information network embedding for recommendation," *IEEE TKDE*, 2018.

[13] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling, "Modeling relational data with graph convolutional networks," in *ESWC*, 2018.

[14] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu, "Heterogeneous graph attention network," in *WWW*, 2019.

[15] Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King, "Magnn: metapath aggregated graph neural network for heterogeneous graph embedding," in *WWW*, 2020.

[16] Zeren Shui and George Karypis, "Heterogeneous molecular graph neural networks for predicting molecule properties," *ICDM*, 2020.

[17] Vassilis N Ioannidis, Da Zheng, and George Karypis, "Panrep: Universal node embeddings for heterogeneous graphs," *arXiv*, 2020.

[18] Yuxiang Ren, Bo Liu, Chao Huang, Peng Dai, Liefeng Bo, and Jiawei Zhang, "Heterogeneous deep graph infomax," *arXiv*, 2019.

[19] Chanyoung Park, Donghyun Kim, Jiawei Han, and Hwanjo Yu, "Unsupervised attributed multiplex network embedding," in *AAAI*, 2020.

[20] Baoyu Jing, Chanyoung Park, and Hanghang Tong, "Hdmi: High-order deep multiplex infomax," in *WWW*, 2021.

[21] Xiao Wang, Nian Liu, Hui Han, and Chuan Shi, "Self-supervised heterogeneous graph neural network with co-contrastive learning," in *KDD*, 2021.

[22] Yanqiao Zhu, Yichen Xu, Hejie Cui, Carl Yang, Qiang Liu, and Shu Wu, "Structure-enhanced heterogeneous graph contrastive learning," in *SDM*, 2022.

[23] Kaveh Hassani and Amir Hosein Khasahmadi, "Contrastive multi-view representation learning on graphs," in *ICML*, 2020.

[24] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio, "Learning deep representations by mutual information estimation and maximization," *ICLR*, 2019.

[25] Yizhou Sun and Jiawei Han, "Mining heterogeneous information networks: a structural analysis approach," *ACM SIGKDD*, 2013.

[26] Yujie Mo, Yuhuan Chen, Liang Peng, Xiaoshuang Shi, and Xiaofeng Zhu, "Simple self-supervised multiplex graph representation learning," in *ACM Multimedia*, 2022.

[27] Qiang Wang, Hao Jiang, Ying Jiang, Shuwen Yi, Qi Nie, and Geng Zhang, "Multiplex network infomax: Multiplex network embedding via information fusion," *Digital Communications and Networks*, 2022.

[28] Thomas N Kipf and Max Welling, "Variational graph auto-encoders," *NIPS Workshop on Bayesian Deep Learning*, 2016.