SPDGI: Meta-Structure and Meta-Path based Deep Graph Infomax

1st Fujiao Ji
College of Computer Science and Engineering,
Shandong University of Science and Technology
Qingdao, China
fujiaoji@sdust.edu.cn

2nd Zhongying Zhao
College of Computer Science and Engineering,
Shandong University of Science and Technology
Qingdao, China
zyzhao@sdust.edu.cn

3rd Chao Li
College of Electronic and Information Engineering,
Shandong University of Science and Technology
Qingdao, China
lichao@sdust.edu.cn

***Abstract*—Network representation learning aims to learn node representations that preserve both structural and attribute information. Due to the rise of mutual information-based methods, researchers have applied them to network representation learning. However, the most common mutual information-based methods focus on dealing with homogeneous networks. Even if some strategies can handle heterogeneous networks, they still do not make full use of nodes' information, like the meta-structures. Therefore, in this paper, we propose an unsupervised graph neural network model, called Meta-Structure and Meta-Path based Deep Graph Infomax (SPDGI) for heterogeneous information network. Specifically, we first employ meta-structure and meta-path to capture semantic information. When dealing with meta-structures, we further divide SPDGI into two ways to capture additional information. We then utilize graph convolution module and semantic level attention mechanism to capture local representations of nodes. Finally, we get the global representation for the graph through an averaging operator and learn the final node representations by maximizing the local-global mutual information. The experimental results on three real-world data sets demonstrate that the proposed SPDGI can achieve good performance.**

***Index Terms*—network representation learning, heterogeneous network, mutual information**

I. Introduction

The goal of network representation learning is to learn the latent and low-dimensional node representations, which preserve the network's topology, vertex content, and other side information [1]. After obtaining representations of nodes, the following tasks (e.g., node classification [2], link prediction [3], recommendation [4]) can be easily and efficiently carried out by applying conventional machine learning algorithms.

Since random walk-based objectives over-emphasize proximity information at the expense of structural information [5], Belghazi *et al.* [6] offer a general-purpose parametric neural estimator of mutual information based on dual representations of the KL-divergence [7]. It is scalable, flexible, and completely can be trainable via back-propagation. Their work attracts significant attention to mutual information-based methods. For example, Deep InfoMax (DIM) [8], Deep Graph Infomax (DGI) [9], Heterogeneous Deep Graph Infomax (HDGI) [10], and DMGI [11]. Although above mentioned mutual information-based algorithms have made great progress, there are still some limitations to be explored. For instance, DIM only focuses on image data; DGI is designed to embed a single network in which only one type of node and edge appear; although HDGI leverages meta-paths to represent the composite relations with different semantics, they still lose some important information, such as when nodes satisfy multiple paths simultaneously; DMGI uses a consensus regularization framework to deal with diverse relationships in multiplex networks. However, the relation type they used is similar to meta-paths and thus has the same disadvantages as HDGI. We summarize related methods in Table I.

TABLE I: The application scenarios and used methods

Methods	DIM	DGI	HDGI	DMGI	SPDGI
Image	✓	×	×	×	×
Homogeneous Network	×	✓	×	×	×
Heterogeneous Network	×	×	✓	×	✓
Multiplex Network	×	×	×	✓	×
Meta-path/Relation	×	×	✓	✓	✓
Meta-strcture	×	×	×	×	✓
Attention	×	×	✓	✓	✓
Mutual Information	✓	✓	✓	✓	✓

To address the aforementioned limitations, we propose a meta-Structure and meta-Path based Deep Graph Infomax (SPDGI) method for heterogeneous information networks (HIN). First, we utilize both meta-structure and meta-path to capture graph heterogeneity rather than meta-paths alone, which allows us to incorporate more complex semantic information. Second, we obtain negative graphs by shuffling node features. Then, we acquire the node's local representations through an attention mechanism on the embedding learned from various meta-structures and meta-paths. Combined nodes' embedding serves as a global representation. Finally, we maximize the mutual information between local node embedding and global graph embedding to deal with the unsupervised settings. According to the different treatment of the meta-structures, we further divide SPDGI into SPDGI-A and SPDGI-P. Both of them tend to utilize the nodes that satisfy diverse paths in meta-structures. SPDGI-A tends to combine these paths, while SPDGI-P is likely to choose nodes that meet all paths. In summary, our contributions are summarized as follows:

- We propose a heterogeneous network representation learning model called SPDGI, which integrates meta-structures, meta-paths, and mutual information in an appropriate way.
- We further divide SPDGI into two approaches, SPDGI-A and SPDGI-P, inspired by the series connection and parallel connection when dealing with meta-structures: confining nodes that satisfy all paths at the same time or any path in meta-structures.
- We conduct extensive experiments to evaluate the performance of our model. Results demonstrate that the representations learned by proposed models are effective for both node classification and clustering tasks. Moreover, we find the applicable conditions of SPDGI-A and SPDGI-P by analyzing the results.

The remainder of this paper is organized as follows. Section II introduces the related works. Section III describes notations used in the paper and presents some preliminary knowledge. We present the SPDGI methodology in Section IV. Experimental evaluations and detailed analyses are discussed in Section V. Finally, the conclusions are presented in Section VI.

II. Related Work

Mutual information is based on Shannon entropy to measure dependence between random variables. Specifically, the mutual information $I(A;B)$ between variable A and B can be understood as the decrease of uncertainty in A given B, just shown as Eq. 1:

$$I(A;B) = H(A) - H(A|B), \tag{1}$$

where H is the Shannon entropy, $H(A|B)$ is the conditional entropy of B given A. Detailed background information are discussed in [6].

However, it is difficult to get mutual information when the probability distributions are unknown. For more general problems, Belghazi *et al.* [6] propose a general-purpose mutual information neural estimator based on dual representations of the KL-divergence [7]. Based on the work of Belghazi *et al.*, Hjelm *et al.* [8] find that depending on the downstream task and maximizing mutual information between the complete input and the encoder output is insufficient for learning efficient representations. Therefore, they introduce DIM to learn representations in the image area, which trains a model to maximize the mutual information between global representations and patches. Although these mutual information-based methods are useful, they are not appropriate for graphs. How to apply mutual information to graphs becomes a difficult but interesting problem. Under these circumstances, Velickovic *et al.* [9] successfully apply it into graphs by maximizing mutual information between patch representations and the corresponding high-level summary of the graph. But the disadvantage is that the proposed model is not suitable for heterogeneous networks, while they are common in the real world. To solve this problem, Ren *et al.* [10] further apply it into heterogeneous networks and propose HDGI. To be specific, they use meta-paths, graph convolution module, and semantic-level attention mechanism to capture individual node's local representations. Considering the deficiency that the above strategies only contain relevant information regarding each relation type, and therefore fail to take advantage of the diversity of networks, Park *et al.* [11] present an unsupervised method for embedding attributed multiplex network, which utilizes a consensus regularization framework and a universal discriminator to jointly integrate the embedding from multiple types of relations between nodes. Detailed differences refer to Table I.

III. Problem Definition

In this section, we first introduce preliminary knowledge. Then, we give the problem definition and summarize the symbols used in this paper in Table II.

Definition 1. Heterogeneous Information Network (HIN) [12–14]. A heterogeneous information network is a directed graph $G = (V, E)$ with a node mapping function $\phi : V \to \mathcal{A}$ and an edge mapping function $\varphi : E \to \mathcal{R}$, where each node $v \in V$ belongs to one node type $\phi(v) \in \mathcal{A}$, and each edge $e \in E$ belongs to a particular relation $\phi(e) \in \mathcal{R}$, respectively. Besides, the type of nodes $|\mathcal{A}|$ and the type of edges $|\mathcal{R}|$ satisfy that $|\mathcal{A}| + |\mathcal{R}| > 2$.

Definition 2. Network Schema [12]. Given a HIN $G = (V, E)$ with a node mapping function $\phi : V \to \mathcal{A}$ and an edge mapping function $\varphi : E \to \mathcal{R}$, its schema T_G is a directed graph defined over node types $\mathcal{A}$ and edge types $\mathcal{R}$, denoted as $T_G = (\mathcal{A}, \mathcal{R})$. For example, Figure. 1(a) is a HIN in DBLP data set. Then, the abstracted network schema is shown as Fig. 1(b)

Definition 3. Meta-path [12]. A meta-path P is defined on the graph of network schema $T_G = (\mathcal{A}, \mathcal{R})$, which is denoted in the form of $\mathcal{A}_1 \xrightarrow{\mathcal{R}_1} \mathcal{A}_2 \xrightarrow{\mathcal{R}_2} \cdots \xrightarrow{\mathcal{R}_l} \mathcal{A}_{l+1}$. It can be defined with a composite edge $\mathcal{R}_1 \circ \mathcal{R}_2 \circ \cdots \circ \mathcal{R}_{l+1}$ between type $\mathcal{A}_1$ and $\mathcal{A}_{l+1}$, where $\circ$ denotes the composition operator on edges.

Definition 4. Meta-structure [15]. Given a HIN $G = (V, E)$ with network schema $T_G = (\mathcal{A}, \mathcal{R})$, a meta-structure S is defined as $S = (\mathcal{A}, \mathcal{R}, v_s, v_t)$, where v_s is the source node, v_t is the target node.

Problem Definition. Meta-structure and meta-path based HIN embedding. Given a HIN $G = (V, E)$ with meta-structures $S = (\mathcal{A}, \mathcal{R}, v_s, v_t)$ and meta-paths P as input, the task is to

TABLE II: Notations and Explanations

Notations	Explanations
G	Graph
E	Edge set
V	Node set
ϕ	Node mapping function
φ	Edge mapping function
$\mathcal{A}$	Type of nodes
$\mathcal{R}$	Type of edges
P	Meta-paths
S	Meta-structure
A	Commuting matrix
$\hat{A}$	Added self-connections commuting matrix
$\tilde{A}$	Commuting matrix of negative examples
M	Meta-paths and meta-structures
K	P The number of M
X	Feature
$\tilde{X}$	Shuffled features
H	Node representations
$\tilde{H}$	Node representations of negative examples
$\mathcal{E}$	Encoder
$\tilde{s}$	Summary vector
$\mathcal{D}$	Discriminator
I_N	Identity matrix
W	Adjacency matrix
W^{M_k}	Layer-specific trainable weight matrix

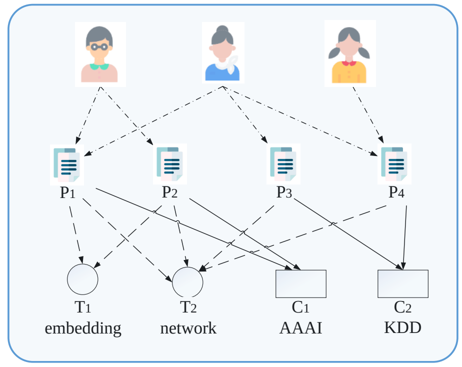
learn the d-dimensional latent representations H for nodes, which not only contains structural and attribute information, but also includes additional but not redundant semantic information.

IV. SPDGI Methodology

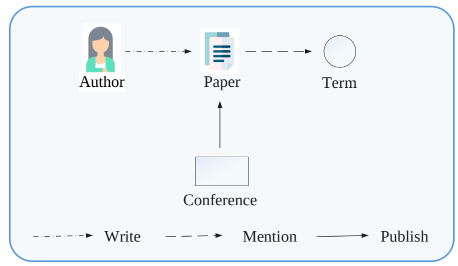
In this section, we propose an unsupervised graph neural network model for heterogeneous information networks. Our model integrates meta-structures, meta-paths, and mutual information appropriately. Figure. 2 shows the whole framework of SPDGI by taking the DBLP data set as an example.

A. SPDGI-A and SPDGI-P

According to the different treatment of meta-structures and meta-paths, we divide SPDGI into SPDGI-A and SPDGI-P. Specifically, SPDGI-A allows nodes at intersections to satisfy any path. It is equivalent to separate the meta-structure into meta-paths, and then combine them into one merged graph. We implement it by adding the commuting matrix. Considering redundancy, we utilize meta-paths that are not included in meta-structures and the merged graph. For SPDGI-P, we constrain nodes to meet all paths. It means that the intersected



(a) HIN example in DBLP data set.



(b) HIN schema in DBLP data set.

Fig. 1: HIN example and abstracted schema in DBLP data set.

TABLE III: Algorithm of SPDGI

Algorithm 1 The overall process of SPDGI.

Input: A HIN graph $G = (V, E)$ with selected meta-structures $S = (\mathcal{A}, \mathcal{R}, v_s, v_t)$ and meta-paths P, initial features X and commuting matrix A.
Output: node representations.
1: Calculate meta-path and meta-structure based commuting matrices A^{M_k} through SPDGI-A and SPDGI-P;
2: Obtain shuffled features $\tilde{X} = Shuffle(X)$;
3: **while** not converged **do**
4: **for** $k = 1...K$ **do**
5: Obtain H^{M_k} and $\tilde{H}^{M_k}$ through Eq. 2;
6: Generate nodes' representations H by Eq. 3;
7: Get a global vector $\vec{s}$ for the whole graph via Eq. 4;
8: Maximize the mutual information with the binary cross-entropy loss of the discriminator through Eq. 5;
9: **end while**

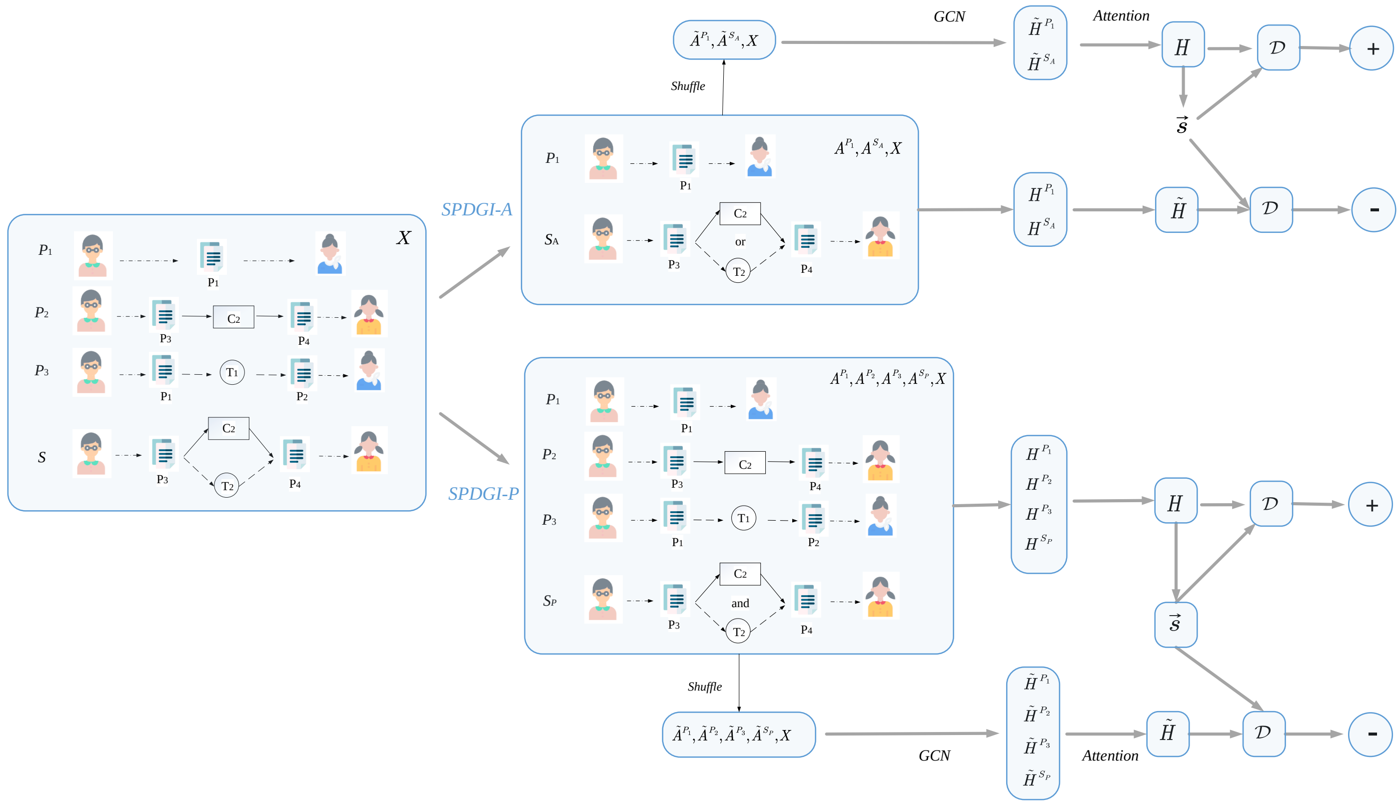


Fig. 2: The overall framework of the proposed SPDGI (Taking DBLP data set as an example)

TABLE IV: Computing commuting matrix of SPDGI-A

SPDGI-A: Commuting matrix
$A^{P_1} = W_{AP} \cdot W_{AP}^{\top}$
$A^{S_A} = W_{AP} \cdot \left(W_{PC} \cdot W_{PC}^{\top} + W_{PT} \cdot W_{PT}^{\top}\right) \cdot W_{AP}^{\top}$

TABLE V: Computing commuting matrix of SPDGI-P

SPDGI-P: Commuting matrix
$A^{P_1} = W_{AP} \cdot W_{AP}^{\top}$
$A^{P_2} = W_{AP} \cdot W_{PC} \cdot W_{PC}^{\top} \cdot W_{AP}^{\top}$
$A^{P_3} = W_{AP} \cdot W_{PT} \cdot W_{PT}^{\top} \cdot W_{AP}^{\top}$
$A^{S_P} = W_{AP} \cdot \left[\left(W_{PC} \cdot W_{PC}^{\top}\right) \odot \left(W_{PT} \cdot W_{PT}^{\top}\right)\right] \cdot W_{AP}^{\top}$

nodes can be reserved when they satisfy all paths in meta-structure. We carry out it via element product between their commuting matrices. The final matrix is combined with meta-path based commuting matrices to obtain the representations. For example, in Fig. 1, there are four types of nodes (Author, Paper, Conference, Term) and three kinds of edges (A-P, P-C, P-T). Selected meta-paths and meta-structures refer to the left side of the Figure. 2. SPDGI-A employs P_1 and S and SPDGI-P leverages P_1, P_2, P_3 and S. The calculation of commuting matrix refers to Table. IV and Table. V.

1) Local Representation: SPDGI shuffles the rows of node feature matrix and keep the commuting matrix unchanged to obtain the negative graph, which is in line with previous works, like [10]. For each M_k ($M \subset \{P, S\}$, k in $[1, K]$), we utilize Graph Convolutional Network (GCN) as our encoder $\mathcal{E}$ because of Ren *et al.*'s work [10]. Therefore, the node representation for each M_k can be obtained by GCN through Eq. 2.

$$H^{M_k} = \left(\hat{D}^{M_k \; -\frac{1}{2}} \hat{A}^{M_k} \hat{D}^{M_k \; -\frac{1}{2}}\right) X W^{M_k}, \tag{2}$$

where $\hat{A}^{M_k} = A^{M_k} + I_N$, I_N is the identity matrix, $\hat{D}^{M_k}$ is the diagonal node degree matrix of A^{M_k} and W^{M^k} is a layer-specific trainable weight matrix.

Then, we add a semantic attention layer to obtain the final node's representation by combining the learned representation, which is consistent with HDGI [10]:

$$H = SemanticLevelAttention\left(\left\{H^{M_k}\right\}_1^K\right). \tag{3}$$

2) Global Representation: The learning objective of SPDGI is to maximize the mutual information between local representations and the global representation. The local node

TABLE VI: Statistics of Experimental Datasets.

Dataset	Node	Edge	Meta-structure	Average Degree (target node)	Class
IMDB	M [4275] A [5431] D [2082] K [7313]	M-A [12838] M-D [4280] M-K [20529]	MAM MDM MKM M(ADK)M	5.15 18.21 78.00 \	3
DBLP	A [4057] P [14328] C [20] T [8789]	A-P [19645] P-C [14328] P-T [88420]	APA APCPA APTPA AP(CT)PA	2.74 1232.56 1669.28 \	4
ACM	P [3025] A [5835] S [56]	P-A [9744] P-S [3025]	PAP PSP P(AS)P	9.68 730.83 \	3

representations are obtained in Section. IV-A1, and we need the summary vector to represent the global information of the entire heterogeneous graph. In this paper, we apply the mean of node representations to get the global summary vector:

$$\vec{s} = \mathcal{R}(H) = \sigma\left(\frac{1}{N}\sum_{i=1}^{N}\vec{h}_i\right). \tag{4}$$

3) Mutual information based discriminator: Inspired by previous works (e.g. [8–11]), we maximize the mutual information based on the Jensen Shannon divergence between the joint and the product of marginals and use the following objectives:

$$\mathcal{L} = \frac{1}{N+M}\left(\sum_{i=1}^{N}\mathbb{E}_{(X,A)}\left[\log\mathcal{D}\left(\vec{h}_i,\vec{s}\right)\right] + \sum_{j=1}^{M}\mathbb{E}_{(\tilde{X},\tilde{A})}\left[1-\log\mathcal{D}\left(\vec{\tilde{h}}_j,\vec{s}\right)\right]\right), \tag{5}$$

where N and M denote the number of nodes and negative examples.

V. Experiment

A. Datasets

To make fair comparisons with HDGI [10], which is the most relevant baseline method, we conduct experiments on the datasets used in their original paper [10] in terms of node classification and node clustering tasks.

- **IMDB.** It contains 4275 movies (M), 5431 actors (A), 2082 directors (D), and 7313 keywords (K). We set movies as the target nodes. For the IMDB dataset, the classification task is to classify movies into three classes (Action, Comedy, and Drama) according to their genre.
- **DBLP.** This is a research paper set, which contains 4057 authors (A), 14328 papers (P), 20 conferences (C), and 8789 terms (T). We set authors as the target nodes. For the DBLP dataset, the classification task is to classify authors into four areas (Database, Data Mining, Information Retrieval, and Machine Learning) according to the research topic.
- **ACM.** It is a research paper set, which contains 3025 papers (P), 5835 authors (A), and 56 subjects (S). For the ACM data set, the classification task is to classify the papers into three classes (Database, Wireless Communication, and Data Mining).

B. Baselines

We compare with some state-of-art baselines to verify the effectiveness of the proposed model.

- **DGI [9]:** It is an unsupervised manner for homogeneous graph, which relies on maximizing mutual information between patch representations and corresponding high-level summaries of the graph. In this paper, we apply DGI to meta-path based homogeneous graph. We first calculate the embeddings for every type, then average embeddings as the final embeddings, and report the final performance.
- **DMGI [11]:** It is an unsupervised network embedding method for the attributed multiplex network, which jointly integrates the node embeddings from multiple graphs by introducing the consensus regularization framework and the universal discriminator.
- **HDGI-C [10]:** It employs meta-paths and graph convolution module with a semantic-level attention mechanism to capture local representations of nodes in heterogeneous information networks. Then, HDGI learns high-level node representations by maximizing the local-global mutual information.
- **SPDGI-A:** The proposed meta-structure and meta-path based deep graph infomax method, which allows nodes to satisfy any path in meta-structures.
- **SPDGI-P:** The proposed meta-structure and meta-path based deep graph infomax method, which constrains nodes to satisfy all paths in meta-structures.

C. Node Classification

We conduct experiments with different training ratios for these three data sets for better comparison. We take fixed 10 percent of data as the validation set. Except for the training data and validation data, the rest data are set as test data. All data are chosen randomly. The dimension of node representations is set as 512 and 256 for SPDGI-A and SPDGI-P, respectively. We set learning rate as 0.0005 after many experiments. The dimension of the semantic-level attention vector is set to 16. And we use early stopping with the patience of 20, i.e. we stop training if the validation loss does not decrease for 20 consecutive epochs. Detailed descriptions refer to parameter analysis. To keep the results stable, we repeat the classification process 10 times and report the average Macro-F1 and Micro-F1 in Table. VII.

Based on Table. VII, we can see that SPDGI has a good performance. For homogeneous graph embedding methods, we apply them to meta-path based homogeneous graphs. Compared with DGI, the proposed SPDGI has a better result in these three data sets because its effectiveness of capturing semantic information. For heterogeneous graph embedding methods, we can observe that DMGI has a good outcome in the ACM data set, while HDGI-C performs better in the DBLP data set. This is because both of them seize heterogeneous information and leverage the strength of mutual information. However, the proposed methods work effectively in most

TABLE VII: Qantitative results on the node classification task.

Dataset	Metrics	Training	DGI	DMGI	HDGI-C	SPDGI-A	SPDGI-P
IMDB	Macro-F1	20%	0.4830	0.6015	0.5844	**0.6698**	0.6055
		60%	0.4894	0.6395	0.6420	**0.6946**	0.6476
	Micro-F1	20%	0.5021	0.6010	0.5846	**0.6712**	0.6024
		60%	0.5048	0.6353	0.6414	**0.6955**	0.6434
DBLP	Macro-F1	20%	0.7379	0.8225	**0.9287**	0.8929	0.9194
		50%	0.7307	0.8400	0.9218	0.9101	**0.9300**
	Micro-F1	20%	0.7478	0.8303	**0.9335**	0.8968	0.9234
		50%	0.7427	0.8477	0.9280	0.9142	**0.9350**
ACM	Macro-F1	20%	0.7322	0.9294	0.9313	**0.9398**	0.9084
		40%	0.7180	0.9280	0.9412	**0.9518**	0.9232
	Micro-F1	20%	0.7670	0.9298	0.9306	**0.9393**	0.9068
		40%	0.7534	0.9278	0.9410	**0.9509**	0.9229

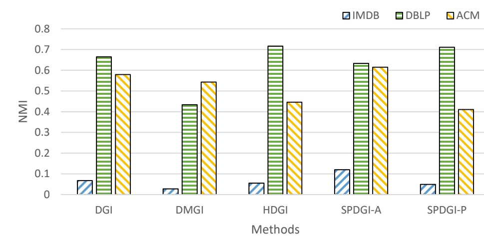
instances. This is due to the properties of data sets and models. SPDGI tends to preserves additional information when nodes satisfy a meta-structure. Particularly, SPDGI-P tends to constrain nodes satisfying each channel in meta-structure, while SPDGI-A is inclined to allow nodes to satisfy any path. Therefore, SPDGI-A is to combine paths in meta-structure, while SPDGI-P is to increase additional information. In the IMDB data set, the degree of nodes is small, thus extra information is able to improve results. In the DBLP data set, the meta-path contains enough information, superfluous information will decrease the performance. Under these circumstances, SPDGI-A performs worse than SPDGI-P and HDGI-C. In ACM data set, the degree distribution of nodes in 'PAP' and 'PAP' meta-paths varies greatly, so capturing nodes that satisfy different paths at the same time may even decay results.

Through the above analysis, we can find that the proposed SPDGI-A and SPDGI-P can achieve good performances when networks contain little information. When there is enough information, SPDGI-P is appropriate to provide additional but not redundant information. When the nodes' distribution varies greatly in meta-structure, SPDGI-A can make a good balance.

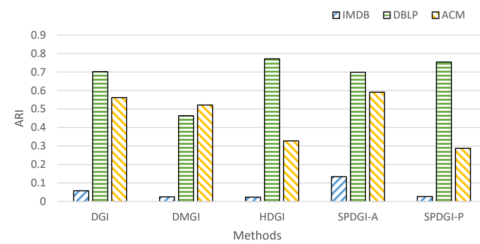
D. Node Clustering.

We also conduct the clustering task to evaluate the embedding learned from the previously mentioned algorithms. Once the proposed SPDGI has been trained, we can get the node embedding. Here we utilize the KMeans to conduct node clustering. The number of clusters of IMDB, DBLP, and ACM data set is set to 5, 4, and 5, respectively. We choose the number of clusters through Elbow Method [16]. We adopt NMI and ARI to assess the quality of clustering results. Since the performance of KMeans is affected by initial centroids, we repeat the process 100 times. Other parameters are same to Section V-C. The average results are reported in Fig. 3.

Form Fig. 3, we can see that DGI cannot perform well in both IMDB and ACM data sets because it is not able to balance weights from various meta-path based homogeneous graphs. However, in the DBLP data set, the DGI model has good results because there is no need to take extra information into consideration while the DBLP already has abundant information. Although HDGI and DMGI are designed for heterogeneous networks, they are still missing some information between nodes that satisfies several paths, making the representations not effective enough. The verification based on node clustering tasks also demonstrates that SPDGI can learn effective representations by considering the additional information. Similar to the classification task, SPDGI-A performs better in IMDB and ACM data sets, HDGI and SPDGI-P can have a good performance in the DBLP data set.



(a) NMI.



(b) ARI.

Fig. 3: Evaluation results on the node clustering task.

E. Parameter Analysis.

In this section, we investigate the sensitivity of parameters, including dimension of the final embedding, dimension of semantic-level attention vector, number of clusters in node clustering task,

Dimension of the final embedding. We investigate the effect of dimension of the final embedding in SPDGI-A. The result in IMDB data set is shown as Fig. 4. We can observe that with the growth of the embedding dimension, the performance raises first and then starts to decrease. The reason is that the proposed models need suitable dimensions to represent information. Morever, smaller or larger dimension may cause difficient representations or additional redundancies. Therefore, considering the performance of result and operating efficiency, we choose 512 and 256 as the embedding dimension for SPDGI-A and SPDGI-P, respectively.

Dimension of semantic-level attention vector. We explore the experimental results with various dimensions of the semantic attention vector. The result is shown in Fig. 5. We can find that SPDGI-A achieves the best performance when the dimension is set to 16. The oversized dimension may lead to the performance starts to degenerate because of overfitting.

VI. Conclusion

In this paper, we propose a simple yet effective unsupervised method for heterogeneous information network representation learning, named SPDGI. It integrates several meta-paths and

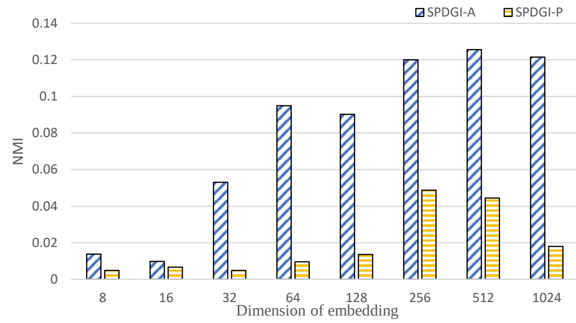
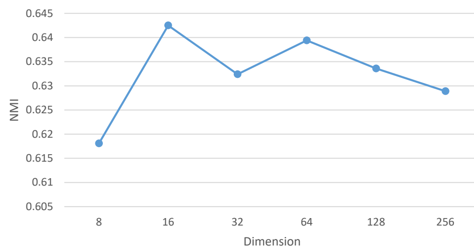
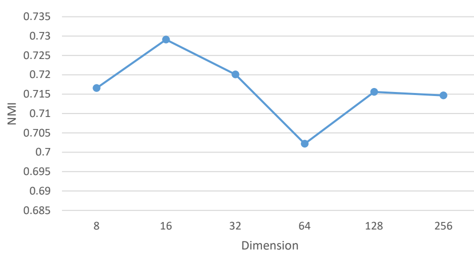


Fig. 4: Dimension of the final embedding.



(a) SPDGI-A



(b) SPDGI-P

Fig. 5: Dimension of the semantic-level attention vector.

meta-structures through an attention mechanism to obtain local representations of nodes and get the global vector by summarizing them. Finally, through maximizing the local-global mutual information, SPDGI learns high-level representations of nodes. We demonstrate the effectiveness of learned representations for both node classification and clustering tasks on three data sets. We are also optimistic that mutual information maximization is a promising future direction for unsupervised representation learning.

Acknowledgment

XXXXX

References

[1] D. Zhang, J. Yin, X. Zhu, and C. Zhang, "Network representation learning: A survey," *IEEE Transaction on Big Data*, vol. 6, pp. 3–28, 2020.

[2] N. Sheikh, Z. T. Kefato, and A. Montresor, "Semi-supervised heterogeneous information network embedding for node classification using 1d-cnn," in *International Conference on Social Networks Analysis, Management and Security*, 2018, pp. 177–181.

[3] T. Li, J. Zhang, P. S. Yu, Y. Zhang, and Y. Yan, "Deep dynamic network embedding for link prediction," *IEEE Access*, vol. 6, pp. 29 219–29 230, 2018.

[4] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu, "Heterogeneous information network embedding for recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, pp. 357–370, 2019.

[5] L. F. R. Ribeiro, P. H. P. Saverese, and D. R. Figueiredo, "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 385–394.

[6] M. I. Belghazi, A. Baratin, S. Rajeswar, S. Ozair, Y. Bengio, R. D. Hjelm, and A. C. Courville, "Mutual information neural estimation," in *Proceedings of the 35th International Conference on Machine Learning*, 2018, pp. 530–539.

[7] A. Ruderman, M. D. Reid, D. García-García, and J. Petterson, "Tighter variational representations of f-divergences via restriction to probability measures," in *Proceedings of the 29th International Conference on Machine Learning*, 2012.

[8] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *7th International Conference on Learning Representations*, 2019.

[9] P. Velickovic, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, "Deep graph infomax," in *International Conference on Learning Representations*, 2019.

[10] Y. Ren, B. Liu, C. Huang, P. Dai, L. Bo, and J. Zhang, "Heterogeneous deep graph infomax," *arXiv preprint arXiv:1911.08538v2*, 2019.

[11] C. Park, D. Kim, J. Han, and H. Yu, "Unsupervised attributed multiplex network embedding," in *34th AAAI Conference on Artificial Intelligence*, 2020, pp. 5371–5378.

[12] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, pp. 992–1003, 2011.

[13] Y. Sun and J. Han, "Mining heterogeneous information networks: a structural analysis approach," *SIGKDD Explorations*, vol. 14, pp. 20–28, 2012.

[14] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *The World Wide Web Conference*, 2019, pp. 2022–2032.

[15] Z. Huang, Y. Zheng, R. Cheng, Y. Sun, N. Mamoulis, and X. Li, "Meta structure: Computing relevance in large heterogeneous information networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1595–1604.

[16] R. L. Thorndike, "Who belongs in the family?" *Psychometrika*, vol. 18, no. 4, pp. 267–276, dec 1953.