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 which preserves both structural and attributed 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 are used to deal with homogeneous networks. Even if there are some methods are able to handle heterogeneous networks, they still not make full use of nodes' information which satisfies several meta-structures simultaneously. 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 network representation learning. Specifically, we first employ meta-structures and meta-paths to capture semantic information. When dealing with meta-structures, we further divide SPDGI into two ways to capture these nodes that satisfy multiple channels at the same time. We then utilize graph convolution module and semantic level attention mechanism to capture node-level representations. Finally, we obtain the summary vector through a readout function and learn the 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 representations of nodes, which preserves network 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.

Rather than relying on random walk-based objectives (e.g. [5–7]), which suffer from over-emphasizing proximity information at the expense of structural information [8], Belghazi *et al.* [9] offer a general-purpose parametric neural estimator of mutual information based on dual representations of the KL-divergence [10], which is scalable, flexible, and completely trainable via back-prop. Due to the work of Belghazi *et al.*, mutual information based methods attract a great number of attentions. For example, Deep InfoMax (DIM) [11], Deep Graph Infomax (DGI) [12], Heterogeneous Deep Graph Infomax (HDGI) [13], and DMGI [14]. Although the mentioned mutual information-based learning algorithms have made a great progress, there are still some limitations need to be explored. For example, DIM focuses on image data; DGI is designed to embed a single network in which only one type of node and edge are appeared; although HDGI leverages meta-paths to represent the composite relations with different semantics, they still lose some important information, like nodes that satisfy multiple paths simultaneously; DMGI is designed for attributed multiplex networks, which uses a consensus regularization framework to deal with diverse relationships. However, the relation type they used is similar to meta-paths and thus also lose information of nodes that satisfying several paths. We summarize the application scenarios and methods used of related models in Table I.

To address the aforementioned limitations, in this paper, we present a meta-structure and meta-path based unsupervised graph neural network model for heterogeneous networks, named Meta-Structure and Meta-Path based Deep Graph Infomax (SPDGI). SPDGI handles graph heterogeneity and obtains nodes' local representations by leveraging meta-structures, gets the final embeddings through an attention mechanism on various meta-structures, and deals with the unsupervised settings by applying mutual information maximization. According to the different treatment of the meta-structures, SPDGI can be further divided into SPDGI-A and SPDGI-P. Both of them tend to utilize nodes that satisfy diverse paths in meta-structure. SPDGI-A tends to combine these paths, while SPDGI-P is likely to choose nodes that meet all paths.

The contributions of this work 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.

TABLE I: The application scenarios and used methods

Methods	Image	Homogeneous Network	Heterogeneous Network	Multiplex Network	Meta-path or Relation	Meta-structure	Mutual Information
DIM	✓						✓
DGI		✓					✓
HDGI			✓		✓		✓
DMGI				✓	✓		✓
SPDGI			✓			✓	✓

- 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 satisfying all paths at the same time or any path.
- We conduct extensive experiments to evaluate the performance of the model. Results demonstrate that the representations learned by proposed models are effective for both node classification and clustering tasks. Morever, the applicable conditions of SPDGI-A and SPDGI-P can be well found 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 introduce SPDGI methodology in detail 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 [9].

However, it is difficult to compute mutual information while the probability distributions are unknown. For more general problems, Belghazi *et al.* [9] propose a general-purpose mutual information neural estimator, which trains a statistics network as a classifier of samples coming from the joint distribution of two random variables and their product of marginals. Based on the work of Belghazi *et al.*, Hjelm *et al.* [11] find that depending on the downstream task and maximizing mutual information between the complete input and the encoder output is insufficient for learning useful representations. Therefore, they introduce DIM to learn representations in 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 become a very interesting but difficult problem. Under the circumstances, Velickovic *et al.* [12] successfully apply it into graph by maximizing mutual information between patch representations and corresponding high-level summaries of graphs. But the disadvantage is that the proposed model is not suitable for heterogeneous networks, while they are common in real world. In order to solve this problem, Ren *et al.* [13] further apply it into heterogeneous networks and proposed HDGI. To be specific, they use meta-paths, graph convolution module and semantic-level attention mechanism to capture individual node local representations. Considering the deficiency that some methods only contained relevant information regarding each relation type, and therefore failed to take advantage of the diversity of networks, Park *et al.* [14] 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. Differences between methods are shown in Table I.

III. Problem Definition

In this section, we first introduce some preliminary knowledge. We then give the problem definition to be solved in this paper. In addition, the symbols used in this paper are summarized in Table II.

Definition 1. Heterogeneous Information Network (HIN) [15–17]. 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. In addition, the types of nodes $|\mathcal{A}|$ and the types of edges $|\mathcal{R}|$ satisfy that $|\mathcal{A}| + |\mathcal{R}| > 2$.

Definition 2. HIN Schema [15]. 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 $T_G = (\mathcal{A}, \mathcal{R})$.

Definition 3. Meta-path [15]. 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 [18]. Given a HIN $G = (V, E)$ with HIN 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 learn the d-dimensional latent representations X for nodes, which not only contains structural and attributed information, but also includes some additional semantic but not redundant information.

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	Adjacency matrix
$\hat{A}$	Added self-connections adjacency matrix
$\tilde{A}$	Adjacency matrix of negative examples
M	Meta-paths and meta-structures
K	P The number of M
X	Feature
$\tilde{X}$	negative examples of X
H	Node representations
$\tilde{H}$	Node representations of negative examples
R	****change Readout function
$\mathcal{C}$	Corruption function
$\mathcal{E}$	Encoder
$\tilde{s}$	Summary vector
$\mathcal{D}$	Discriminator
I_N	Identity matrix
W	Commuting matrix
W^{M_k}	Layer-specific trainable weight matrix

IV. SPDGI Methodology

In this section, we first make a brief illustration of the proposed model by taking DBLP data set as an example in Section IV-A, illustrated in Fig. 1. Second, we further divide SPDGI into two methods based on the framework of SPDGI, which both provide additional information when dealing with meta-structures and meta-paths. A detailed description is presented in Section IV-B. Third, we obtain node-level representations by utilizing graph convolutional network and semantic level attention in Section IV-C. Finally, we employ readout function and a mutual information based discriminator to get the global-level representation in Section IV-D.

A. SPDGI Architecture Overview

1) Calculate the meta-path and meta-structure based adjacency matrices A^{M_k} with respect to SPDGI-A and SPDGI-P for k in $[1, K]$, where A is the adjacency matrix of target nodes, M can be P or S, and K is the number of M;
2) Use corruption function $\mathcal{C}$ to obtain the negative examples:
$$\left(\tilde{X}, \tilde{A}^{M_k}\right) \sim \mathcal{C}\left(X, A^{M_k}\right), \tag{2}$$
where X represents features of nodes;
3) Obtain local representations $\vec{h}_i^{M_k}$ and the negative local representation $\tilde{\vec{h}}_i^{M_k}$ for M_k:
$$H^{M_k} = \mathcal{E}\left(X, A\right), \tilde{H}^{M_k} = \mathcal{E}\left(\tilde{X}, \tilde{A}^{M_k}\right), \tag{3}$$
where H^{M_k} and $\tilde{H}^{M_k}$ is local representations of nodes and negative nodes in M_k, $\mathcal{E}$ is the encoder;
4) Generate node representations through an attention mechanism:
$$H = attention\left(\left\{H^{M_k}\right\}_1^K\right), \tilde{H} = attention\left(\left\{\tilde{H}^{M_k}\right\}_1^K\right), \tag{4}$$
where H and $\tilde{H}$ denote graph-level nodes and negative graph-level nodes;
5) Get a summary vector via a readout function:
$$\vec{s} = \mathcal{R}\left(H\right), \tag{5}$$
where $\vec{s}$ is the whole graph representation, $\mathcal{R}$ is the readout function;
6) Train the discriminator $\mathcal{D}$ to maximize the mutual information between positive nodes and the summary vector $\vec{s}$.

B. SPDGI-A and SPDGI-P

The main difference between SPDGI-A and SPDGI-P lies on how to deal with the meta-structures, especially the computing of commuting matrix. Taking DBLP data set as an example, there are four type of nodes (Author, Paper, Conference, Term) and three kind of edges (A-P, P-C, P-T). Selected meta-paths and meta-structures can refer to Fig. 1. When dealing with meta-structure S_4, we can allow nodes to satisfy any path, or we constrain nodes to meet all channels. By analyzing the process of these two strategies, we propose SPDGI-A to utilize nodes that satisfy any path in meta-structure, which tends to combine every channel, while we present SPDGI-P to leverage nodes that meet all channels, which can bring extra information to the network. In DBLP data set, SPDGI-A employs P_1 and S_4, and the calculation of commuting matrix refers to Table. III. SPDGI-P leverages P_1, P_2, P_3 and S_4. In addition, the calculation of commuting matrix can refer to Table. IV, which is different from SPDGI-A.

TABLE III: Computing commuting matrix of SPDGI-A

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

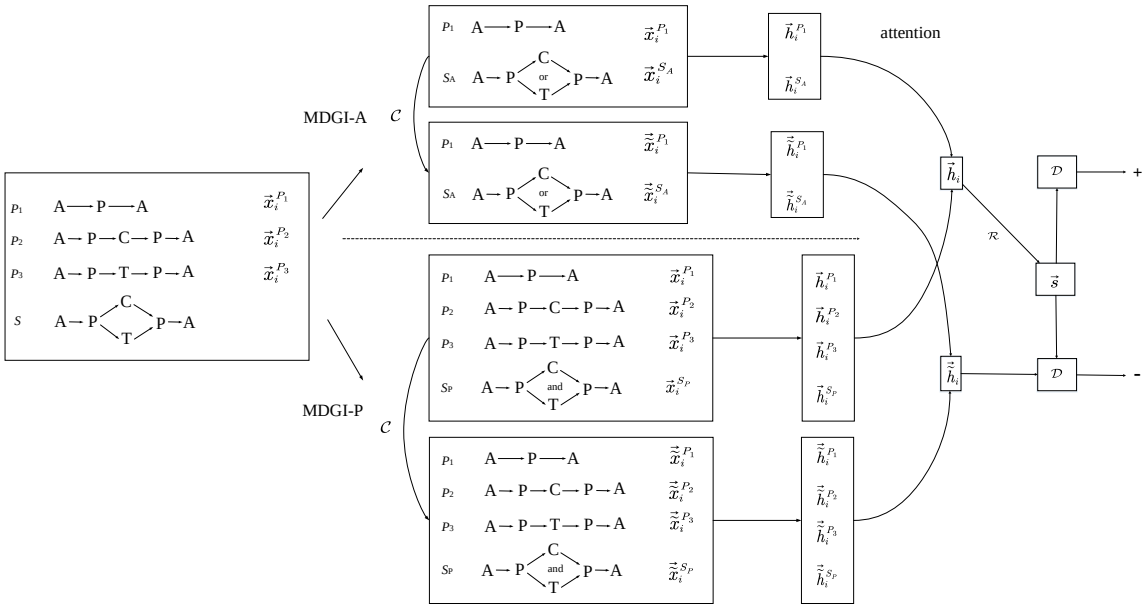


Fig. 1: The overall framework of the proposed SPDGI in DBLP data set

TABLE IV: Computing commuting matrix of SPDGI-P

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

C. Node-level Representation

For each M_k (k in $[1, K]$), we utilize Graph Convolutional Network (GCN) as our encoder $\mathcal{E}$ because of Ren *et al.*'s work [13]. Therefore, the node representation for each M_k can be obtained by GCN through Eq. 6.

$$H^{M_k} = \left(\hat{D}^{M_k\ -\frac{1}{2}} \hat{A}^{M_k} \hat{D}^{M_k\ -\frac{1}{2}} \right) X W^{M_k}, \tag{6}$$

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 learn the weights, which is consistent with HDGI [13]:

$$H = attention \left(\left\{ H^{M_k} \right\}_1^K \right). \tag{7}$$

The representations H can be further leveraged in the global representation of the graph-level summary which will be described in Section. IV-D.

D. Global-level Representation

1) Readout function: The learning objective of SPDGI is to maximize the mutual information between local representations and the global representation. The local representations of nodes are obtain in Section. IV-C, and we need the summary vector to represent the global information of the entire heterogeneous graph. In this paper, we apply the averaging operator as our readout function, where we simply take the mean of node representations to output the graph-level summary vector:

$$\vec{s} = \mathcal{R}(H) = \sigma \left(\frac{1}{N} \sum_{i=1}^{N} \vec{h}_i \right). \tag{8}$$

2) Mutual information based discriminator: Inspired by previous works (e.g. [11–14]), we maximize the mutual information based on the Jensen Shannon divergence between the joint and the product of marginals and use the following objectives:

$$\mathcal{L} = \frac{1}{N+M} \left(\sum_{i=1}^{N} \mathbb{E}_{(X,A)} \left[\log \mathcal{D} \left(\vec{h}_i, \vec{s} \right) \right] + \sum_{j=1}^{M} \mathbb{E}_{(\tilde{X},\tilde{A})} \left[1 - \log \mathcal{D} \left(\vec{\tilde{h}}_j, \vec{\tilde{s}} \right) \right] \right), \tag{9}$$

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

3) Corruption function: In this paper, we also keep the adjacency matrices unchanged and shuffle the rows of node feature matrix as the corruption function, which is in line with previous works, like [13].

V. Experiment

A. Datasets

To make fair comparisons with HDGI [13], which is the most relevant baseline method, we conduct experiments on the datasets used in their original paper [13] 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

TABLE V: 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

movies as the target nodes. For 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 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 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 verfify the effectiveness of the proposed model.

- **DGI [12]:** It is an unsupervised manner for homogeneous graph, which relies on maximizing mutual information between patch representations and corresponding high-level summaries of 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 [14]:** It is an unsupervised network embedding method for attributed multiplex network, which jointly integrates the node embeddings from multiple graphs by introducing the consensus regularization framework and the universal discriminator.
- **HDGI-C [13]:** 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 based heterogeneous deep graph infomax method, which allows nodes to satisfy any path.
- **SPDGI-P:** The proposed meta-graph based heterogeneous deep graph infomax method, which constrains nodes to satisfy all paths.

C. Implementation Details

For the proposed SPDGI, we set the learning rate to 0.0005, dimensions of node-level representations are set as 512 and 256 for SPDGI-A and SPDGI-P, the dimension of semantic-level attention vector to 16. Detailed reasons can refer to Section V-F. And we use early stopping with a patience of 20, i.e. we stop training if the validation loss does not decrease for 20 consecutive epochs. We use Pytorch to implement our model and conduct experiments in the CPU.

D. 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 validation set. Except for the training data and validation data, the rest data are set as test data. All data are choosed randomly. Detailed descriptions refer to parameter analysis. To keep the results stable, we repeat the classification process for 10 times and report the average Macro-F1 and Micro-F1 in Table. VI.

Based on Table. VI, we can see that SPDGI has a good performance. For homodegeneous graph embedding methods, we apply them to meta-path based homogeneous graph. Compared with DGI, the proposed SPDGI has a better result in these three data sets because it captures rich semantic information. For heterogeneous graph embedding methods, we can observe that DMGI has a good outcome in ACM data set, while HDGI-C performs better in DBLP data set. This is because both of them seize heterogeneous information. However, the proposed methods works effectiely in most of instances. This is due to the properties of data sets and models. SPDGI-P tends to constrain those nodes which satisfy 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 addtional information. In IMDB data set, the degree of nodes is small, thus extra information is able to improve results. In DBLP data set, meta path contains enough information, superflous informaion will decrease the performance. Under this curcumstances, 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 contains little information. When there are enough information, SPDGI-P is appropriate to provide additional but not redundant information. When the nodes' distribution vary greatly in meta-structure, SPDGI-A can make a good balance.

E. Node Clustering.

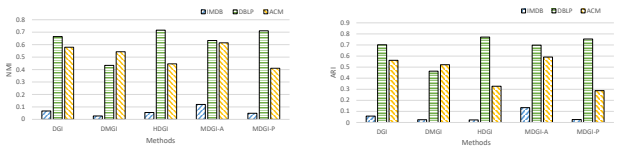
We also conduct the clustering task to evaluate the embeddings learned from the abovementioned algorithms. Once the proposed SPDGI has been trained, we can get the node embedding via feed forward. Here we utilize the KMeans to conduct node clustering. The number of clusters of IMDB,

TABLE VI: 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

DBLP and ACM data set are set to 5, 4, and 5, respectiely. Detailed information refer to parameter analysis in Section V-F. 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 for 100 times and report the average results in Fig. 2.

DGI cannot perform well in both IMDB and ACM data sets, because it is not able to handle the graph heterogeneity, while it has a good performance in DBLP data set due to abundant information provided by meta-path based homogeneous network. Although HDGI and DMGI are designed for heterogeneous networks, they are still missing some information between nodes which 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. Similarly to classification task, SPDGI-A performs better in IMDB and ACM data sets, HDGI and SPDGI-P can have a good performance in DBLP data set.



(a) NMI. (b) ARI.

Fig. 2: Evaluation results on the node clustering task.

F. 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. The result in IMDB data set is shown as Fig. 3. 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.

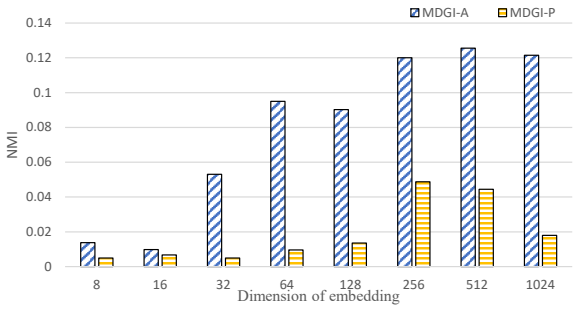
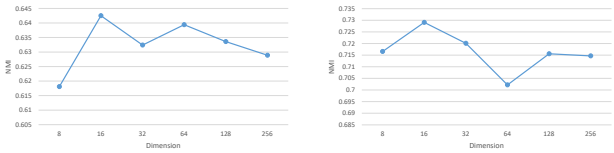


Fig. 3: Dimension of the final embedding.

Dimension of semantic-level attention vector. We explore the experimental results with various dimensions of semantic attention vector. The result is shown in Fig. 4. We can find that SPDGI achieves the best performance when the dimension is set to 16. Oversized dimension may lead to the performance starts to degenerate because of overfitting.



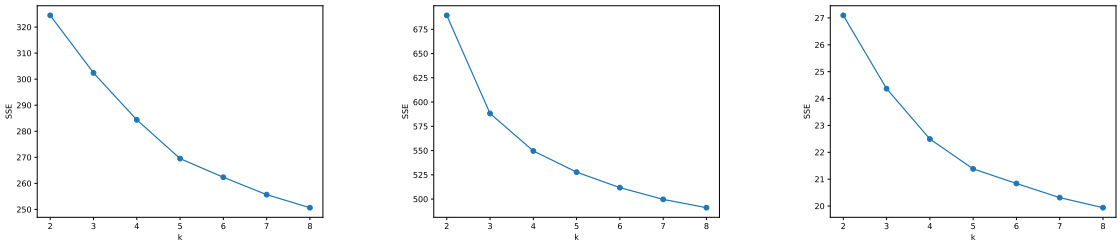
(a) SPDGI-A (b) SPDGI-P

Fig. 4: Dimension of the semantic-level attention vector.

Number of cluters in node clustering task. In order to check the impact of cluters in node clutering task, we utilize Elbow Method [19] to choose the appropriate number of clusters. We apply SPDGI-A to three data sets and take sum of squared errors as criterion. From Fig. 5, we can see that the clusters can be choosen as 5, 4 and 5 for IMDB, DBLP and ACM data sets, respectively.

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 meta-structures through an attention mechanism to obtain local representations of nodes and get the summary vector by utilizing a readout function. Finally, through maximizing the local-global mutual information, SPDGI learns high-level representations. 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.



(a) IMDB. (b) DBLP. (c) ACM.

Fig. 5: Parameter sensitivity of SPDGI-A to the clusters in node clustering task.

ACKNOWLEDGMENT

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] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: online learning of social representations," in *International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701–710.

[6] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: large-scale information network embedding," in *International Conference on World Wide Web*, 2015, pp. 1067–1077.

[7] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 855–864.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

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

[14] 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.

[15] 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.

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

[17] 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.

[18] 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.

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