MMAN: Metapath Based Multi-Level Graph Attention Networks for Heterogeneous Network Embedding (Student Abstract)

Jie Liu,[1,2] Lingyun Song,[1,2] Li Gao,[1,2] Xuequn Shang,[1,2]

[1]School of Computer Science, Northwestern Polytechnical University, Xi'an China

[2]Key Laboratory of Big Data Storage and Management, Ministry of Industry and Information Technology, Northwestern Polytechnical University, Xi'an China

{jayliu@mail.nwpu.edu.cn, lysong@nwpu.edu.cn, lig@mail.nwpu.edu.cn, shang@nwpu.edu.cn, }

Abstract

Current Heterogeneous Network Embedding (HNE) models can be roughly divided into two types, i.e., relation-aware and metapath-aware models. However, they either fail to represent the non-pairwise relations in heterogeneous graph, or only capable of capturing local information around target node. In this paper, we propose a metapath based multi-level graph attention networks (MMAN) to jointly learn node embeddings on two substructures, i.e., metapath based graphs and hypergraphs extracted from original heterogeneous graph. Extensive experiments on three benchmark datasets for node classification and node clustering demonstrate the superiority of MMAN over the state-of-the-art works.

Introduction

Heterogeneous Network Embedding (HNE) has been a challenging task due to multiple vertice and relation types and diverse feature spaces of node content. Current HNE models can be roughly divided into two categories: *relation-aware* and *metapath-aware* methods. Relation-aware methods(Zhang et al. 2019; Hu et al. 2020) directly aggregate information from neighboring nodes, with attention mechanism which assigns different weights for different relations. However, due to the hop constraint, the information they capture is somewhat local. Metapath-aware methods(Wang et al. 2019; Fu et al. 2020) use metapath, a composite relation between two vertices, to transform heterogeneous graph into multiple homogeneous graphs, which can be then learned by homogeneous GNNs. However, in most cases, metapaths simultaneously connect over two end nodes, causing these models lose semantic integrity.

To address the limitations above, we introduce a novel **metapath based Multi-level Graph Attention Networks**, namely, **MMAN**. MMAN first constructs metapath-based graph and hypergraph extracted from original heterogeneous graph and then hierarchically conducts graph-level and hypergraph-level aggregation to generate more comprehensive node embeddings. We evaluate the proposed method for node classification and node clustering on heterogeneous graphs. Experiment results show the superiority of our model over the state-of-the-art works.

Copyright © 2022, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Our Proposed Approach

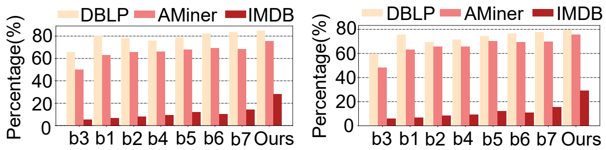
Unifying Content Feature Space In order to project all types of node features into the same latent space, we have $\mathbf{x}'_i = \mathbf{W}_b \cdot \mathbf{x}_i$, where $\mathbf{x}_i \in \mathbb{R}^{d_b}$ and $\mathbf{x}'_i \in \mathbb{R}^d$ are the former and transformed features of node i, respectively. $\mathbf{W}_b \in \mathbb{R}^{d_b \times d}$ is the linear transformation matrix exclusively for nodes of type $b \in \mathcal{B}_v$.

Graph-level Aggregation Given a metapath based graph $\mathcal{G}_s^{\mathcal{P}} = \{\mathcal{V}_s, \mathcal{E}_s^{\mathcal{P}}\}$ extracted from original heterogeneous graph G, we first perform the aggregation on $\mathcal{G}_s^{\mathcal{P}}$ to preliminarily learn the local pairwise relations and obtain the basic embeddings of the end nodes. Since each metapath based neighbor of certain node shows different importance in aggregation, we introduce attention mechanism to assign weight coefficients to different neighbors. The generated embeddings $\mathbf{x}^*$ will be further processed in the hypergraph-level attention.

Hypergraph-level Aggregation For metapath based hypergraph $\mathcal{G}_h^{\mathcal{P}} = \{\mathcal{V}_h, \mathcal{E}_h^{\mathcal{P}}\}$, hypergraph-level aggregation is composed of two components: *intra hyperedge aggregation* and *inter hyperedge aggregation*. Intra hyperedge aggregation combine the information of nodes within the same hyperedge and incorporate it with intermediate path feature to compute a single vector (metapath based hyperedge embedding). Let $\mathbf{x}_i^*$ be the i-th node vector, $e_j \in \mathcal{E}_i$ indicates all the hyperedges connected to it, $v_k \in e_j$ is the node within hyperedge e_j and h_{τ_j} is intermediate path feature. We have $\mathbf{h}_j = \mathcal{F}_{intra}(\mathbf{h}_{\tau_j}, \{\mathbf{x}_k^*, \forall v_k \in e_j\})$, where $\mathcal{F}_{intra}$ indicates intra hyperedge aggregation.

Inter hyperedge aggregation generates the target node embedding by aggregating information of all the metapath based hyperedges connected to it. We have $\mathbf{f}_i^{\mathcal{P}} = \mathcal{F}_{inter}(\mathbf{x}_i^*, \mathbf{h}_j, \forall e_j \in \mathcal{E}_i\})$, where $\mathcal{F}_{inter}$ indicates inter hyperedge aggregation. Attention mechanisms are applied at both steps of intra and inter hypergraph aggregation. Then, MMAN concats all the metapath specific node embeddings and generate final node embeddings $\mathbf{f}_i = \mathbf{f}_i^{\mathcal{P}_1} \oplus \cdots \oplus \mathbf{f}_i^{\mathcal{P}_c}$.

Semi-supervised and Unsupervised Training Through the multi-level graph aggregation schema above, we have obtained the low-dimension embeddings for all the nodes in heterogeneous graph. For semi-supervised learning, with the guide of a small fraction of labeled nodes, we apply cross entropy between the output prediction and labels as the loss



(a) ARI (b) NMI

Figure 1: Quantitative results on node clustering task.

function, formulated as:

$$\mathcal{L} = -\frac{1}{|\mathcal{B}_v|}\sum_{b\in\mathcal{B}_v}\frac{1}{|\mathcal{V}_b|}\sum_{i\in\mathcal{V}_b} y_i \mathbf{log}(\widetilde{y}_i), \tag{1}$$

where $\mathcal{B}_v$ is node type set, y_i is the one-hot label vector of node i and $\widetilde{y}_i$ is the predicted probability vector generated by Multi-Layer Perceptron (MLP):$\widetilde{y}_i = MLP(\mathbf{f}_i)$.

For unsupervised learning, we propose a hyperedge negative sampling loss function as followed:

$$\mathcal{L} = -\sum_{e\in\mathcal{E}}\sigma(\sum_{\mu\in e}\text{log}\mathbf{f}_\mu) - \sum_{e'\in\mathcal{E}^-}\sigma(\sum_{\mu'\in e'}-\text{log}\mathbf{f}_{\mu'}). \tag{2}$$

where $\mathcal{E}$ is the set of observed hyperedges, while $\mathcal{E}^-$ is the set of negative hyperedges sampled from unobserved hyperedges.

Experiments

Dataset and Baseline Algorithms We evaluate our proposed model on three real-world datasets:*DBLP*, *AMiner* and *IMDB*. We compare MMAN with seven state-of-the-art embedding methods including two *homogeneous graph embedding*, i.e. GCN(Kipf and Welling 2017) and GAT(Veličković et al. 2018), one *heterogeneous hypergraph embedding* method, i.e., DHNE(Tu et al. 2018) and four *heterogeneous graph embedding* methods, i.e., HAN(Wang et al. 2019), MAGNN(Fu et al. 2020), HGT(Hu et al. 2020) and HetGNN(Zhang et al. 2019).

Parameter Settings MMAN is trained for 100 epochs with early stopping strategy. The graph attention and hypergraph attention component both consist of one layer with hidden units set to 128 and 64, respectively. We set learning rate to 0.005 for DBLP and IMDB and 0.01 for AMiner, respectively. The number of attention head T is 8 and dropout rate is 0.5. We utilize L2 regularization to avert overfitting and set weight decay to 0.001. We split 20% nodes as training set, 10% as validation set and others as test set for all datasets. For baseline models, we optimize their hyperparameters with validation sets, separately.

Node Classification Performance As Table 1 shows, MMAN outperforms all the baselines on all three evaluation datasets, which demonstrates the superiority of our method on node classification task. On IMDB and AMiner, MMAN outperforms the second to best baseline HAN by 4%~12%, which demonstrates the rich information gain in embedding process provided by the structure and feature content embedded in intermediate path. Compared with the most competitive baseline MAGNN, MMAN has 4%~6% improvement over it, which strongly proves the effectiveness to concern multiple relations when encoding metapath instances. As for DBLP, MMAN outperforms the best baseline MAGNN by 1%~2%.

Dataset Metrics(%)	DBLP Mic-F1	DBLP Mac-F1	AMiner Mic-F1	AMiner Mac-F1	IMDB Mic-F1	IMDB Mac-F1
GCN (b1)	90.15	89.56	90.23	90.48	49.78	45.73
GAT (b2)	91.56	91.11	90.78	91.56	55.28	49.44
DHNE (b3)	79.46	78.84	80.68	79.55	42.23	43.64
HetGNN (b4)	88.10	87.95	89.83	90.06	50.12	51.56
HGT (b5)	91.00	90.82	93.59	92.97	58.43	57.79
HAN (b6)	92.33	91.69	92.74	92.68	56.73	54.03
MAGNN (b7)	92.15	92.20	93.24	92.67	59.85	59.43
MMAN	**93.37**	**93.15**	**96.86**	**96.62**	**60.42**	**60.53**

Table 1: Experiment results on node classification task

Node Clustering Performance We also conduct node clustering task. We extract the latent embeddings of labeled nodes from trained models and feed them into K-Means algorithm. *Normalized mutual information* (NMI) and *adjusted rand index* (ARI) are used as the evaluation metrics. The results are showed in Figure 1. It's clear to see that MMAN consistently outperforms the other baselines on all three datasets.

Conclusion

This paper proposes a metapath based multi-level graph attention networks (MMAN) for heterogeneous graph embedding. The experiment results on node classification and clustering tasks demonstrate the superiority of MMAN over seven state-of-the-art algorithms.

Acknowledgements

This work is supported by The National Key Research and Development Program of China (Grant 2020AAA0108504).

References

Fu, X.; Zhang, J.; Meng, Z.; and King, I. 2020. MAGNN: metapath aggregated graph neural network for heterogeneous graph embedding. In *WWW*, 2331–2341.

Hu, Z.; Dong, Y.; Wang, K.; and Sun, Y. 2020. Heterogeneous graph transformer. In *WWW*, 2704–2710.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. *International Conference on Learning Representations*.

Tu, K.; Cui, P.; Wang, X.; Wang, F.; and Zhu, W. 2018. Structural deep embedding for hyper-networks. In *AAAI*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2018. Graph attention networks. *ICLR*.

Wang, X.; Ji, H.; Shi, C.; Wang, B.; Ye, Y.; Cui, P.; and Yu, P. S. 2019. Heterogeneous graph attention network. In *The World Wide Web Conference*, 2022–2032.

Zhang, C.; Song, D.; Huang, C.; Swami, A.; and Chawla, N. V. 2019. Heterogeneous graph neural network. In *KDD*, 793–803.