










Multi-view Heterogeneous Network Embedding

Ouxia Du , Yujia Zhang , Xinyue Li , Junyi Zhu , Tanghu Zheng ,
and Ya Li  

School of Computer and Information Science,
Southwest University, Chongqing 400715, China
crystal@swu.edu.cn

Abstract. In the real world, the complex and diverse relations among different objects can be described in the form of networks. At the same time, with the emergence and development of network embedding, it has become an effective tool for processing networked data. However, most existing network embedding methods are designed for single-view networks, which have certain limitations in describing and characterizing the network semantics. Therefore, it motivates us to study the problem of multi-view network embedding. In this paper, we propose a Multi-View Embedding method for Heterogeneous Networks, called MVHNE. It mainly focuses on the preservation of the network structure and the semantics, and we do not process them separately, but consider their mutual dependence instead. Specifically, to simplify heterogeneous networks, a semantics-based multi-view generation approach was explored. Then, based on the generated semantic views, our model has two concerns, namely the preservation of single-view semantics and the enhanced view collaboration. With extensive experiments on three real-world datasets, we confirm the validity of considering the interactions between structure and semantics for multi-view network embedding. Experiments further demonstrate that our proposed method outperforms the existing state-of-the-art methods.

Keywords: Heterogeneous network · Multi-view network · Enhanced view collaboration · Network analysis · Network embedding

1 Introduction

Networks are ubiquitous in the real world, such as social networks, academic collaboration networks, citation networks, etc. In recent years, network embedding technology has gradually become an important network analysis tool to study networked data [3, 5, 11, 13, 18]. The main idea of network embedding is mapping the network to a low-dimensional latent space to maximize the preservation of

This work is supported by National Natural Science Foundation of China (No. 61603310), the Fundamental Research Funds for the Central Universities (No. XDJK2018B019).

network topological, as well as node attributes information. The effectiveness of network embedding has been demonstrated in many downstream network learning tasks, such as link prediction [1, 3], node classification [21, 23], and community detection [6].

Depending on the network topology and semantic features, we can categorize two different types of networks, namely, homogeneous and heterogeneous networks [3]. Usually, there is the only same type of nodes and edges in homogeneous networks, such as academic collaboration networks and citation networks. However, homogeneous networks are difficult to describe objects due to their complex interactions in the real world. Thus, the focus of most current embedding work gradually shifts towards heterogeneous network embeddings. The networks are heterogeneous containing different types of nodes and edges. For example, in E-commerce networks, users are connected via goods and stores. Heterogeneous are able to describe richer semantics in nodes and edges than homogeneous networks. In recent years, the research and analysis of heterogeneous networks have become a hot issue in the field of data mining [19]. No matter for homogeneous or heterogeneous networks, most embedding methods are designed for single-view scenarios. The existence of multiple type relations among nodes will be difficult to describe due to the limitations of single network. In addition, the data contained in a single network as a result of sparsity and measurement errors or data access limitations may be noise and incomplete [7]. If the embedding method is applied directly on such data, the final embedding quality will be less ideal. Embedding methods to solve heterogeneous networks from a multi-view perspective have begun to emerge in recent years. The noise or incomplete data contained in one view may be modified or supplemented in another view [14, 22]. It overcomes the limitations brought by the single view.

Inspired by [16], we propose a novel Multi-View Embedding model for Heterogeneous Network, called MVHNE. First, to deal with multi-semantic relations in the heterogeneous network, we generate views based on the preset semantics. Usually, meta-path can be used to capture specific semantic of heterogeneous networks. Then, we mainly focus on semantic preservation in single view and collaborative learning across views. In enhanced view collaboration, the interactions between network structure and semantics are explored. After that, the view embeddings are aggregated by a self-attention mechanism. Finally, the global network topological information is used to enhance the final node representation. In summary, our contributions are summarized as follows:

- We propose a novel multi-view network embedding model, which incorporates view semantics with global network topology.
- We propose a novel and flexible enhanced cross-view collaboration, consisting of First collaboration and Second collaboration. The goal of the first collaboration is to align the same nodes in all the views, while the second collaboration is to align the neighbors.
- We conduct extensive experiments with three real-world datasets, and empirically show that our proposed method performs better than the other baselines.

The rest of this paper is organized as follows. Section 2 introduces the work related to our research. Section 3 formalizes the research problem. Then the details of our proposed embedding method are described in Sect. 4. Section 5 summarizes the experimental results. Finally, the conclusion and discussion are made in Sect. 6.

2 Related Work

Network Embedding. Network embedding has emerged as an efficient approach for generating the node representation for real networks or graphs in recent years. In general, network embedding methods can be divided into two categories: homogeneous network embedding and heterogeneous network embedding. In the study of homogeneous network embedding, many works take preserving network topological structural as priority. By applying word embedding to network embedding, DeepWalk [12], Node2vec [4] and LINE [17] take a random walk-based approach to learn the low-dimensional representations of the nodes. GAT [20] embeds network with neighbors' feature using self-attention strategy. The designs of most recent embedding approaches are based on heterogeneous network [15, 21]. HERec [15] adopts a meta-path based random walk strategy on heterogeneous information network embedding for recommendation. Heterogeneous network embedding focus on how semantic is extracted from network, and most of these designs are closely related to meta-paths. Although these heterogeneous network embedding methods consider network semantics, the mutual influence among different semantics are not explored deeply.

Multi-view Network Embedding. Due to the complexity of relations in the real world, single-view embedding methods present certain limitations. Therefore, multi-view embedding methods for simplified heterogeneous networks based on split ideas are gaining popularity [7, 10, 14, 16]. Multi-view network embedding aims to capture the multiple relations that exist in the network. [14] studies multi-view network representation learning, which proposes to use attention mechanism to learn the weights of different views to vote for robust node representations. [16] summarizes the two key points of preservation and collaboration in multi-view embedding. [2] applies the multi-view approach on the heterogeneous network and proposes a transformation and inductive model to learn node representation. There are a lot of works involving the topic of multi-view network embeddings, but almost none of them study view collaboration under network structure and semantic interaction.

3 Problem Definition

In this section, for clarity, some important notations are summarized in Table 1. Then we give some definitions relevant to this work as well as the problem to be addressed.

Definition 1 (Heterogeneous network): A heterogeneous network is denoted as $G = (U, E, W)$, consisting of a nodes set U and an edges set E . It is also associated with a node type mapping function $\phi : U \rightarrow A$ and an edge type mapping function $\psi : E \rightarrow R$. A and R correspond to the set of nodes type and edges type, where $|A| + |R| > 2$. $w_{ij} \in W$ is the weight of the edge $e_{ij} \in E$, indicating the strength of the relation between the nodes u_i and u_j .

Table 1. Notations

Notation	Description
G	Heterogeneous information network
V	Set of all network views
U/A	Set of all nodes/nodes types
E/R	Set of all edges/edges types
W	Weight of the edges
Ω	Set of node pairs generated based on random walk
d	The dimension of common/final embedding
m	The dimension of single view embedding

Definition 2 (Multi-view network): A multi-view network can be defined by the $G' = (U, E, V, U_v, E_v)$, where U and E is nodes set and edges set, respectively. V denotes views set, U_v and E_v respectively indicate the set of nodes and edges in a specific view $v \in V$.

Definition 3 (Multi-view Network embedding): Given a multi-view network G' , it goal is learning a mapping function $f : U \rightarrow R^{n \times d}$, where $d \ll |U|$ and f can map the original multi-view network G' to a low-dimensional representation space.

4 Methodology

A multi-view embedding model for heterogeneous network (called MVHNE) is presented in this section. MVHNE consists of three major components, including semantics-based view generation, view preservation and collaboration, as well as embedding fusion. the overall framework is shown in Fig. 1.

4.1 Semantics-Based View Generation

For the efficient processing of the heterogeneous networks, we aim at decomposing the complex relations in the original network. To this end, a semantics-based view generation approach is explored. Usually, meta-path is an important concept used to describe patterns of underlying semantic relations among different types of nodes. Here, we extract views corresponding to specific semantics from

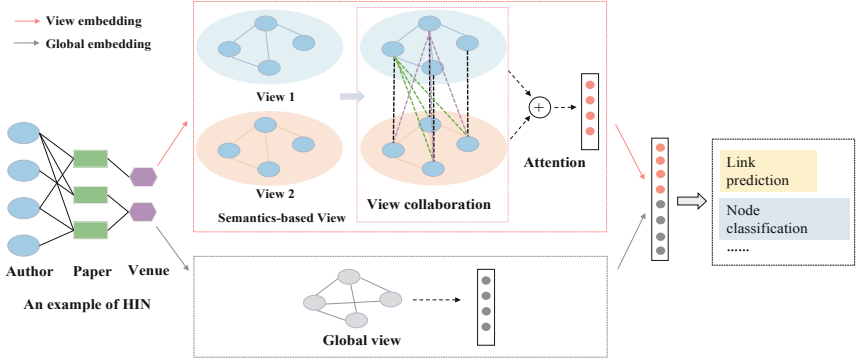


Fig. 1. The framework of our proposed MVHNE model.

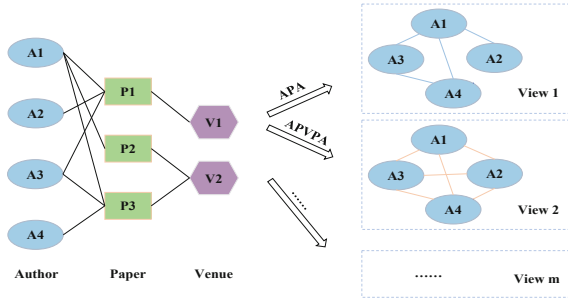


Fig. 2. An illustrative example of the semantics-based view generation in an academic network. Suppose we generate view 1 based on path $P1(A - P - A)$ representing co-author relationships. Under the constraints of the $P1$, we can obtain a sub-network of authors and papers. Then we filtered all paper nodes in this sub-network. Lastly, we treat the remaining part as the final view generated by $P1$.

the original network based on preset meta-paths. And then, we apply constraints and filtering to each view. Constraints and filtering are designed to facilitate processing. Semantic-based view generation is described in detail in a simple example described in Fig. 2.

4.2 View Preservation and Enhanced View Collaboration

Single-View Semantics Preservation. Through the previous section, we know that each view reflects specific network semantics. However, these views are decoupled from each other, so it is necessary to preserve the different relations among nodes separately. Following [2], we define the node u_i 's k -th ($0 \leq k \leq K$) level embedding $u_{i,v}^{(k)}$ in view v as the mean aggregation of neighbors' embeddings, i.e., $u_{i,v}^{(k)} = \sigma(\hat{W} \cdot aggregator(u_{i,v}^{(k-1)}, \forall u_j \in N(u_i, v)))$. $N(u_i, v)$ is the neighbors of node u_i in view v , $u_{i,v}^{(0)}$ is a random initialization embedding, the *aggregator*

function is a mean aggregator. Then, to preserve the diversity of each view, our main task is to model node pairs relations. Specifically, we generated node sequences using random walk, and skip-gram model is used to update node embeddings. Note that node pairs $\Omega^{(v)}$ are formed by central nodes and context nodes in the sequence. Thus, for node pair $(u_i, u_j) \in \Omega^{(v)}$, MVHNE aim to minimize the following negative log-likelihood:

$$L_S = - \sum_{v \in V} \sum_{(u_i, u_j) \in \Omega^{(v)}} p(u_{j,v} | u_{i,v}) \quad (1)$$

Here, the conditional probability $p(u_{j,v} | u_{i,v})$ is defined by the softmax function, i.e., $p(u_{j,v} | u_{i,v}) = \exp(\boldsymbol{\nu}_{i,v} \cdot \boldsymbol{\nu}_{j,v}^{\sim}) / \sum_{n \in U} \exp(\boldsymbol{\nu}_{i,v} \cdot \boldsymbol{\nu}_{n,v}^{\sim})$, where $\boldsymbol{\nu}_{i,v}$ and $\boldsymbol{\nu}_{i,v}^{\sim}$ are the center embedding and context embedding of node u_i , respectively.

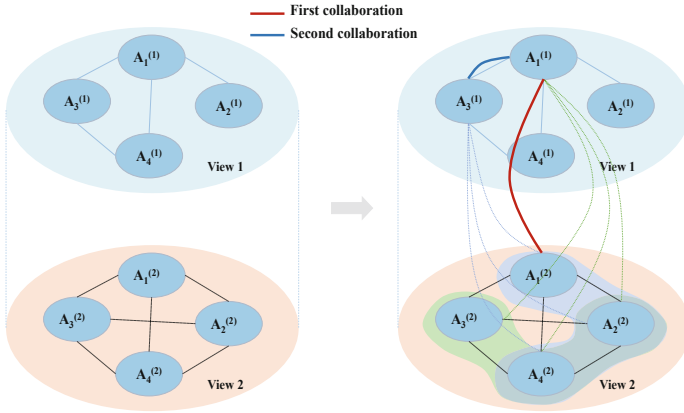


Fig. 3. Enhanced cross-view collaboration.

To improve the training efficiency, Negative sampling [9] is adopted. The objective function can be written as:

$$L_S = -[\log \sigma(\boldsymbol{\nu}_{i,v} \cdot \boldsymbol{\nu}_{j,v}) + \sum_{k=1}^l E_{u_k \sim P_v(u)} \sigma(\boldsymbol{\nu}_{i,v} \cdot \boldsymbol{\nu}_{k,v}')] \quad (2)$$

where l is the number of negative samples, u_k is negative node sampled from the noise distribution $P_v(u) \sim d_u^{3/4}$, and d_u is the degree of the node u , $\sigma(\cdot)$ denotes an activation function. By optimizing L_S , multiple relations among the nodes are preserved.

Enhanced Cross-View Collaboration. We propose an enhanced cross-view collaboration approach, which achieves a mutual complement of semantics in different views. Enhanced view collaboration consists of the first and second collaboration. Two collaborations among views can be interpreted by Fig. 3.

First Collaboration. To capture self-proximity, first collaboration refers to align the same nodes in each view. Specifically, for the view $v, v' \in V$, node $u_{i,v}$ in view v and node $u_{i,v'}$ in view v' should be aligned, such as $A_1^{(1)}$ and $A_1^{(2)}$ in Fig. 3. The first collaboration loss L_{First} is defined as Eq. (3), where $\nu_{i,v'}$ and $\nu_{i,v}$ are the embedding of the node u_i in views v and v' , respectively.

$$\begin{aligned} L_{First} &= -\sum_{v \in V} \sum_{u_i \in U} \sum_{v \neq v'} p(u_{i,v'} | u_{i,v}) \\ &= -[\log \sigma(\nu_{i,v} \cdot \nu_{i,v'})] + \sum_{k=1}^L E_{u_k \sim P_v(u)} \sigma(\nu_{i,v} \cdot \nu_{k,v'}) \end{aligned} \quad (3)$$

Second Collaboration. Inspired by literature of [10], we design the second collaboration with many-to-many ideas. Since the first collaboration does not touch the neighborhood structure of nodes, we design second collaboration for further enhancing the collaboration among views. The main idea of the second collaboration is that for two adjacent nodes in one view, they will also have a similar neighborhood structure in the other. For example, for pair $(A_1^{(1)}, A_3^{(1)})$ in Fig. 3, second collaboration aims to align A_1 's neighbors $\{A_2^{(2)}, A_3^{(2)}, A_4^{(2)}\}$ and A_3 's neighbors $\{A_1^{(2)}, A_2^{(2)}, A_4^{(2)}\}$.

Formally, for node u_i in view v , we define the neighborhood features $\nu_{i,(v \rightarrow v')}$ via its neighbors $u_k \in N(u_i, v')$ in view v' , as shown in Eq. (4). Here, $\nu_{k,v'}$ denotes the embedding of node u_k in view v' , W is edge weight.

$$\nu_{i,(v \rightarrow v')} = \frac{\sum_{u_k \in N(u_i, v')} W_{i,k,v'} \cdot \nu_{k,v'}}{\sum_{u_k \in N(u_i, v')} W_{i,k,v'} + 1} \quad (4)$$

Next, for each pair (u_i, u_j) in view v , we measure the proximity $\nu_{i,v}$ and $\nu_{j,v}$ and their corresponding neighborhood feature $\nu_{i,(v \rightarrow v')}$ and $\nu_{j,(v \rightarrow v')}$, respectively. Then we minimize the disagreement between these two proximity as second collaboration loss L_{Second} .

$$L_{Second} = \sum_{v \in V} \sum_{i,j=1}^{|v|} \sum_{v \neq v'} [\nu_{i,v}(\nu_{j,v})^T - \nu_{i,(v \rightarrow v')}(\nu_{j,(v \rightarrow v')})^T] \quad (5)$$

where the superscript T denotes the transposition of the vector, $|v|$ is the number of nodes in view v . To learn enhanced node embedding, we designed the First and Second collaboration between views. The proposed view collaboration also explores the interaction between network structure and semantics while considering the network alignment.

4.3 Embedding Fusion

Semantic Embedding with View Attention. After the above steps, we can obtain the view embedding for each node. However, considering that the diverse connections in the network are of different importance to the nodes. Therefore, we

introduce the self-attention mechanism proposed in [8] to model the importance of different views. Specifically, we integrate the obtained view representation $\nu_{i,v}$ with *Concatenate()* operation by Eq. (6). Further, the view weight coefficient $a_{i,v}$ is calculated by Eq. (7), where $W_v^{(1)}, W_v^{(2)}, b_v^{(1)}, b_v^{(2)}$ are adjustable parameters.

$$\nu'_i = \text{Concatenate}(\nu_{i,v}), v \in V \quad (6)$$

$$a_{i,v} = \text{softmax}(W_v^{(2)} \tanh(W_v^{(1)} \nu'_i + b_v^{(1)}) + b_v^{(2)}) \quad (7)$$

Global Information Fusion. To not lose the global information of the network, we view the final node embedding as consisting of two parts, i.e., view embedding and common embedding. In particular, we add the common embedding $\nu_{i,g}$ for each node u_i , which is shared among different views. The final embedding ν_i is as follows:

$$\nu_i = \nu_{i,g} + \tanh\left(\sum_{v \in V} a_{i,v} \nu_{i,v}\right) \quad (8)$$

With the above processes, the finally learned node representation not only contains the semantics relations in multiple views but also grasps the global topological information.

4.4 Optimization Objective

In this section, the overall loss function of MVHNE can be summarized by the loss of semantic preservation in single view, the loss of view collaboration and the loss for specific task. As shown in Eq. (9), the above three losses represented by L_S, L_{First}, L_{task} , respectively. λ, μ, η are adjustable hyper-parameters.

$$L = L_S + \lambda L_{First} + \mu L_{Second} + \eta L_{task} \quad (9)$$

We explore the performance of MVHNE on two tasks, i.e., link prediction and node classification. For the link prediction task, we can view it as a bi-classified task. We define loss $L_{task} = L_{lp} = \frac{1}{|U_t|} \sum_{i,j \in U_t} [-y_{ij} \log y_{ij} - (1 - y_{ij}) \log(1 - y_{ij})]$, where y_{ij} indicates whether there is a link between the node i and the node j . If there is an edge, $y_{ij} = 1$, otherwise 0. $\tilde{y}_{ij} = f(\nu_i, \nu_j)$ is the result of model prediction, where f is a classification output function, and $U_t \in U$ is a set of nodes for training. Similarly, for the node classification task, we define loss $L_{task} = L_{nc} = \frac{1}{|U_t|} \sum_{i \in U_t} [-y_i \log y_i - (1 - y_i) \log(1 - y_i)]$, the y_i is the real class of the node i , and the \tilde{y}_i is the result of the model classification prediction. After that, we adopt gradient descent method and the back-propagation method to update and minimize the objective function. When the objective function does not converge, we update the corresponding parameter values in the model by calculating derivative.

Time Complexity. In our method, The time complexity of generating the node view embedding is $O(|E|Sd)$, where $|E| = \sum_{v \in V} |E^{(v)}|$, d is the final node embedding dimension, and S is the number of negative samples. For specific training tasks, such as link prediction, the time complexity is $O(TSd)$, where T is the size of the training dataset. Usually, T is much smaller than E , so the total time complexity of the model is $O(|E|Sd)$.

5 Experiments

5.1 Experimental Setup

Datasets. Three real datasets, ACM, Aminer and DBLP, are used in our experiments. A semantic view of each dataset is generated based on a specific meta-path. More details about the experimental data are shown in Table. 2.

Table 2. Statistics of datasets.

Dataset	# of Nodes	# of Edges	Semantics-based Views
ACM	paper(P):4019 author(A):7167 subject(S):60	17426	PAP PSP
Aminer	paper(P):6564 author(A):13329 reference(R):35890	76838	PAP PRP
DBLP	author(A):4057 paper(P):14328 conference(C):20 term(T):7723	119783	APA APCPA APTPA

Baselines. We compare MVHNE to several state-of-the-art network embedding models for experimental analysis. Note that for the single-view method used for contrast, we first run the model on each view and then report the average performance. DeepWalk [12] adopts a truncated random walk and skip-gram model to generate node embeddings. LINE [17] is a single-view embedding model that preserves the first and second-order similarity of the network. Node2vec [4] exploits a biased random walk and explores the network neighbors through a set of parameters p and q . MNE [22] is a multi-view embedding model. It proposes a high dimensional common embedding and low dimensional additional embedding for each relation type, then learn multiple relations jointly based on a unified embedding model. GATNE [2] is a multiplex heterogeneous network embedding model. Here we only consider the transductive model GATNE-T, because we do not introduce node attribute information. GATNE-T views the overall embedding as consisting of base embedding and edge embedding.

Implementation Details. For all models, we uniformly set the embedding dimension as 32. For DeepWalk and Node2vec, we set walk length as 10, window size as 3, and the number of negative samples as 5. For Node2vec, the high parameter p and q are set as 2 and 0.5. LINE proposes first and second order proximity, in which we set the embedding dimensions of both ways as 16, respectively. The other parameters for baselines are referred from the papers and

tuned to be optimal. We chose the parameter $\lambda = 5$, $\mu = 0.5$, and $\eta = 1000$ for MVHNE, and the number of neighbors in view collaboration is fixed to 10.

For the link prediction task, we adopt ROC-AUC, widely used to measure the quality of the model, which can be obtained by summing the area of the parts under the ROC curve. For node classification task, we employ micro-F score to evaluate the accuracy of the model performance.

5.2 Link Prediction

Link prediction is a task commonly used to evaluate the quality of embedding. We evaluated the link prediction performance of MVHNE on the three datasets using ROC-AUC metrics. In experiments, we describe the link prediction task as a binary classification problem, where negative sample are generated by selecting five nodes without relations randomly. Then, the final embeddings are used to further train a logistic regression model using five-fold cross-validation.

The experimental results of three datasets are shown in Table. 3, we can gain the following conclusions. (1) MVHNE achieves the best performance by comparing with all the state-of-the-art approaches on the link prediction task. The results show that the multi-view embedding model has higher prediction accuracy than the single-view model, it demonstrates that the multiple views can capture more comprehensive relations among nodes. (2) Comparison of MVHNE with other multi-view models shows that MVHNE achieves higher AUC. The reason may be that MNE and GATNE only consider the preservation of single-view semantics and ignoring the view collaboration, while MVHNE touches both simultaneously.

Table 3. Performance of different methods on link prediction.

	ACM		DBLP		Aminer	
	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC	ROC-AUC	PR-AUC
Deepwalk	0.7808	0.6962	0.7632	0.7563	0.7334	0.7321
LINE	0.8014	0.7145	0.7956	0.7821	0.7244	0.7169
Node2vec	0.8335	0.7757	0.8133	0.8025	0.7662	0.7654
MNE	0.8359	0.7648	0.8422	0.8201	0.7865	0.7695
GATNE	0.8471	0.7377	0.8281	0.8057	0.7812	0.7649
MVHNE	0.8509	0.7981	0.8548	0.8492	0.8105	0.7868

5.3 Node Classification

We experimentally analyze the effectiveness of the proposed model on the node classification task. Micro-F is used to evaluate the classification performance on different datasets of the proposed model. Figure 4 illustrates MVHNE outperform other comparison models on classification task. Specifically, DeepWalk,

LINE, Node2vec learn the node representations based on the network structure, which only captures one aspect of semantic among the nodes. By contrast, the multi-view embedding models consider the different network semantics while preserving the network structure. In some way, multi-view methods learning more comprehensive information. At the same time, MVHNE shows its superiority when compared to other multi-view models that do not consider view collaboration. Further analysis, MVHNE focused on view collaboration, mining potential relationship between network structure and semantics to generate more robust node representations.

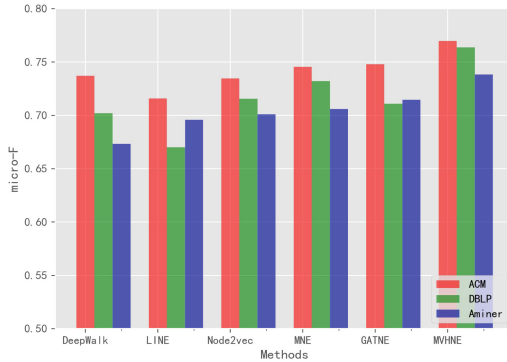


Fig. 4. Performance of different methods on node classification.

5.4 Parameter Sensitivity Analysis

We analyze the sensitivity of λ, μ, η in MVHNE. Since the model has similar performance on each dataset, we take for example the link prediction task on the ACM. λ and μ represent the importance of first collaboration and second collaboration between views, respectively. We take a value of $[0, 0.5, 1, 2, 5, 8]$ for λ and μ , as shown in Fig. 4(a)(b). From the experimental results reflected in Fig. 4(b), the model performs poorly when we do not consider the second collaboration between views (i.e., $\mu = 0$). Combining Figs. 4(a) and 4(b), we can see that the ROC-AUC score rises to the highest when $\lambda = 5$ and $\mu = 0.5$, indicating that the second collaboration between views we considered is very meaningful in enhancing the node representation. In addition, we also analyze the changes of the AUC corresponding to η at $[1, 10, 100, 1000, 10000]$. As shown in Fig. 4(c), MVHNE performs best when $\eta = 1000$.

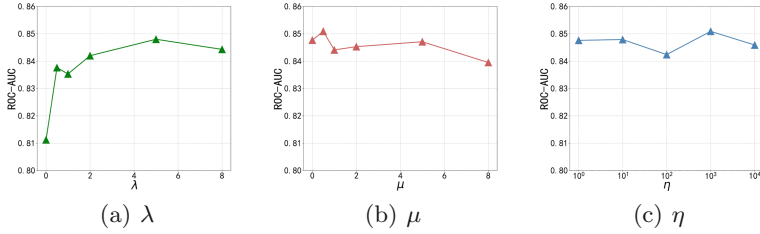


Fig. 5. Influence of the parameter λ, μ, η on ACM dataset.

6 Conclusion

In this paper, we proposed MVHNE to solve the problem of multi-view network embedding. In particular, to obtain the view embedding of node, we proposed First view collaboration and Second view collaboration. The First view collaboration was used to align the same nodes in all views. Furthermore, we have proposed a novel enhanced view collaboration, aiming to align the neighbors of nodes. The interaction between network structure and semantics has been explored through enhanced view collaboration. Extensive experiments were conducted on three real-world datasets, and the empirical results demonstrated that MVHNE outperforms other state-of-art methods on link prediction and node classification tasks. Although the proposed embedding model performs well on different tasks, some node label information needs to be known in advance. In some real-world environments, the acquisition of node label information is very challenging and expensive. Therefore, in future work, we will explore a self-supervised learning method that mines supervised signals from the data itself to learn the representation of nodes on heterogeneous information networks.

References

1. Aziz, F., Gul, H., Muhammad, I., Uddin, I.: Link prediction using node information on local paths. *Phys. A: Statist. Mech. Appl.* **557**, 124980 (2020)
2. Cen, Y., Zou, X., Zhang, J., Yang, H., Zhou, J., Tang, J.: Representation learning for attributed multiplex heterogeneous network. In: *KDD*, pp. 1358–1368 (2019)
3. Cui, P., Wang, X., Pei, J., Zhu, W.: A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* **31**(05), 833–852 (2019)
4. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2016)*, pp. 855–864 (2016)
5. Hao, P., Qing, K., Ceren, B., M., D.R., Yong-Yeol, A.: Neural embeddings of scholarly periodicals reveal complex disciplinary organizations. *Sci. Adv.* (2021)
6. He, D., et al.: Community-centric graph convolutional network for unsupervised community detection. In: Bessiere, C. (ed.) *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pp. 3515–3521 (2020)
7. Kircali, S.A., Yuan, F., Min, W., Jiaqi, S., Keong, C.K., Xiaoli, L.: Multi-view collaborative network embedding. *ACM Trans. Knowl. Discov. Data*, pp. 1–18 (2021)

8. Lin, Z., et al.: A structured self-attentive sentence embedding. In: ICLR (2017)
9. Mnih, A., Teh, W.Y.: A fast and simple algorithm for training neural probabilistic language models. In: ICML, pp. 419–426 (2012)
10. Ni, J., et al.: Co-regularized deep multi-network embedding. In: WWW '18: The Web Conference 2018, Lyon, April 2018, pp. 469–478 (2018)
11. Niu, J., Gao, Y., Qiu, M., Ming, Z.: Selecting proper wireless network interfaces for user experience enhancement with guaranteed probability. *J. Parall. Distrib. Comput.* **72**(12), 1565–1575 (2012)
12. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: online learning of social representations. *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2014)*, pp. 701–710 (2014)
13. Qiu, M., Liu, J., Li, J., Fei, Z., Ming, Z., Sha, E.H.: A novel energy-aware fault tolerance mechanism for wireless sensor networks. In: 2011 IEEE/ACM International Conference on Green Computing and Communications, pp. 56–61 (2011)
14. Qu, M., Tang, J., Shang, J., Ren, X., Zhang, M., Han, J.: An attention-based collaboration framework for multi-view network representation learning. In: CIKM, pp. 1767–1776 (2017)
15. Shi, C., Hu, B., Zhao, W., Yu, P.S.: Heterogeneous information network embedding for recommendation. *IEEE Trans. Knowl. Data Eng.* **31**(02), 357–370 (2019)
16. Shi, Y., Han, F., He, X., Yang, C., Luo, J., Han, J.: mvn2vec: preservation and collaboration in multi-view network embedding. arXiv preprint [arXiv:1801.06597](https://arxiv.org/abs/1801.06597) (2018)
17. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: large-scale information network embedding. In: *Proceedings of the 24th International Conference on World Wide Web*, pp. 1067–1077 (2015)
18. Wang, J., Qiu, M., Guo, B.: Enabling real-time information service on telehealth system over cloud-based big data platform. *J. Syst. Archit.* **72**, 69–79 (2017). *Design Automation for Embedded Ubiquitous Computing Systems*
19. Wang, X., Bo, D., Shi, C., Fan, S., Ye, Y., Yu, P.S.: A survey on heterogeneous graph embedding: methods, techniques, applications and sources. arXiv preprint [arXiv:2011.14867](https://arxiv.org/abs/2011.14867) (2020)
20. Wang, X., et al.: Heterogeneous graph attention network. *The Web Conference on The World Wide Web Conference WWW 2019 (WWW 2019)*, pp. 2022–2032 (2019)
21. Yu, B., Hu, J., Xie, Y., Zhang, C., Tang, Z.: Rich heterogeneous information preserving network representation learning. *Pattern Recogn.* **108**, 107564 (2020)
22. Zhang, H., Qiu, L., Yi, L., Song, Y.: Scalable multiplex network embedding. In: IJCAI, pp. 3082–3088 (2018)
23. Zhang, Z., et al.: Anrl: attributed network representation learning via deep neural networks. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 3155–3161 (2018)