BSOGCN: Brain Storm Optimization Graph Convolutional Networks Based Heterogeneous Information Networks Embedding

Liang Qu, Huaisheng Zhu
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, China
{qul,11611725}@mail.sustech.edu.cn

Yuhui Shi*
Department of Computer Science and Engineering
Southern University of Science and Technology
Shenzhen, China
shiyh@sustech.edu.cn

***Abstract*—Recently, Graph Convolutional Networks (GCNs) have shown great potential in the field of graph embedding. They map the nodes of the graph into the low dimensional vectors by aggregating the neighbor nodes' features information. However, most existing GCNs only focus on the homogeneous information networks instead of the heterogeneous information networks (HINs) with multiple types of nodes which are more common in the real world. Because the different types of neighbor nodes could have different impacts on the target nodes, it is difficult to manually design a proper neighbor nodes' features information aggregating weights. To address this problem, we propose a novel HINs embedding algorithm based on the brain storm optimization (BSO) algorithm and the graph convolutional network (GCN), called BSOGCN, which utilizes BSO to optimize the neighbor nodes' features information aggregating weights. It can be applied to the various HINs under various scenarios without any prior knowledge. The proposed method has been evaluated on both inductive and transductive node multi-class classification tasks on three real-world HINs datasets. The experimental results demonstrate that BSOGCN is competitive against other state-of-the-art methods.**

***Index Terms*—graph embedding, graph convolutional networks, heterogeneous information networks, brain storm optimization.**

I. Introduction

Heterogeneous information networks (HINs) with multiple types of nodes or edges are ubiquitous in many real-world problems [1], [2], such as academic citation networks consisting of paper nodes, author nodes and venue nodes; movie rating networks consisting of movie nodes and people nodes; and business review networks consisting of business nodes and customer nodes. There are many networks information mining tasks aiming to mine valuable information from these HINs, such as node classification [3], node clustering [4], link prediction [5], [6], community detection [7] and so on. To address these problems, recently, many graph embedding algorithms [8] are proposed to map the nodes of the graph into the low dimensional vectors and preserve the graph structural and semantic information as much as possible. The vectors

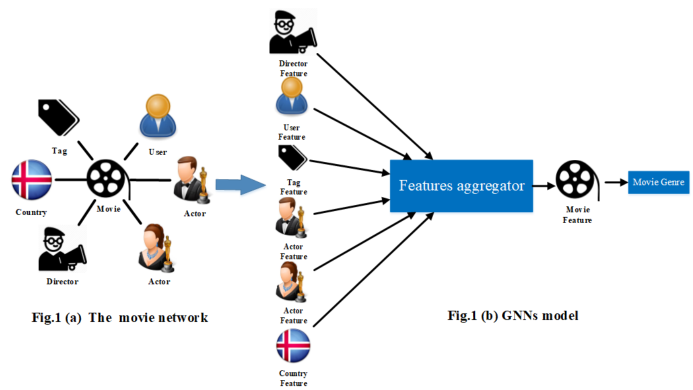

Fig. 1. A simple example that using GNNs model classifies movie genre in a heterogeneous movie network.

can be utilized as the nodes' features, and are input into the off-the-shelf algorithms related to the downstream tasks.

The early graph embedding algorithms are studied as the dimension reduction problem, such as principal component analysis (PCA) [9], multidimensional scaling (MDS) [10], Isomap [11] and local linear embeddings (LLE) [12]. These methods aim to preserve the pairwise node proximity information, however, difficult to generalize to large scale graphs because of expensive computation. Inspired by the success of the neural language model such as Skip-gram [13] on the words representation learning problem, methods like DeepWalk [14], node2vec [15] and metapath2vec [16] are proposed. They use various random walk based node sampling strategies to obtain the paths of the graphs, and derive the node embeddings by the Skip-gram [13]. However, these methods suffer from drawbacks that 1) they only preserve the graph topology information and cannot utilize the graph semantic information such as node attributes and labels; 2) they cannot efficiently obtain the embeddings for the new coming nodes. Recently, to overcome the above problems, the GCNs [17] models are proposed, they obtain the node embeddings by aggregating the local neighbor nodes' features (attributes) information and

* Corresponding author

can directly obtain the new coming node embeddings because of weights sharing mechanism. However, most existing GCNs methods treat the different types of neighbor nodes equally, which is not suitable for the HINs. Because various types of neighbor nodes may have different impact on the target nodes.

For example, as shown in the Figure 1, for a heterogeneous movie information network consisting of of 6 types of nodes (i.e. Movies(M), Users(U), Tags(T), Directors(D), Actors(A) and Countries(C).) and 5 types of edges (M-A, M-C, M-D, M-T and M-U.), the goal is to obtain the movie node's embedding vector and classify its movie genre. Most existing GCNs aggregate 6 types of neighbor nodes with same weights, however, for a node classification task (e.g. predicting the movie genre), intuitively, the tag nodes may have larger impact on movie node than other types of nodes. Thus, it should assign the different weights to the different types of nodes during the neighbor nodes aggregating process.

To address the aforementioned problem, we propose a novel HINs embedding algorithm based on BSO [18] and GCN [3], called BSOGCN, which utilizes BSO to optimize the neighbor nodes aggregating weights.

The contributions of this paper are summarized as follows:

- To address the existing GCNs failing to effectively deal with HINs embedding problems, a novel HINs embedding algorithm BSOGCN is proposed. It has three advantages comparing with the traditional HINs embedding algorithms. **Inductive**: BSOGCN can effectively obtain the new coming node embeddings; **Heterogeneous**: BSOGCN can learn the different types of neighbor nodes aggregating weights for HINs under various scenarios. **Parallel**: BSOGCN is easy to parallelize.
- The proposed algorithm BSOGCN is validated on three real-world HINs datasets for both inductive and transductive node multi-class classification task. The experimental results show that the proposed method is competitive against other state-of-the-art methods.

The rest of the paper is organized as follows: Section 2 introduces the related works; Section 3 provides the basic definitions about the research problem; Section 4 gives a detailed introduction of the proposed method; the experimental setting and results are provided in Section 5 followed by a conclusion and future works in Section 6.

II. Related Work

In this section, we will introduce some related works about the BSOGCN, including the homogeneous information networks embedding methods (i.e. DeepWalk [14], Node2vec [15], GCN [3] and Graph Attention Network (GAT) [19].) and the heterogeneous information networks embedding methods (i.e. metapath2vec [16], HIN2vec [20], and GraphInception [21].)

A. Homogeneous Information Networks Embedding

DeepWalk [14] and Node2vec [15] are the representative homogeneous information networks embedding algorithms, they combine Skip-Gram [13] and random walk based nodes sampling strategies to obtain node embeddings. However, they cannot be applied to the HINs and efficiently deal with the new coming nodes.

GCN [3] and GAT [19] are representative GCNs based homogeneous information networks embedding methods, and they can efficiently obtain the new coming node embeddings. Both of them obtain node embeddings by aggregating neighbor nodes' features. GCN [3] assigns the same aggregating weights to the neighbor nodes' features, which cannot discriminate the importance of different types of nodes in HINs. GAT [19] utilizes the attention mechanism to discriminates the importance of different types of nodes. Similar to BSOGCN, it also learn the aggregating weights of the neighbor nodes' features. However, it utilizes the gradient based optimization method to optimize the weights, which is easy to fall into local optimum.

B. Heterogeneous Information Networks Embedding

To learn the node embeddings in HINs, metapath2vec [16] and HIN2vec [20] utilize the metapaths [2] based node sampling strategies. They sample nodes according to the predefined metapaths, which can simultaneously capture network's structural and semantic information. However, the designs of metapaths highly rely on the human's prior knowledge, which cannot effectively generalize to the various scenarios HINs. Furthermore, they also cannot efficiently deal with the new coming nodes.

GraphInception [21] integrates the metapaths into the GCN to obtain the node embeddings in HINs. It utilizes metapaths to transform HINs to some homogeneous subgraphs and then do a collective node embeddings. However, the designs of metapaths also relies on the human's prior knowledge.

III. Preliminaries

In this section, we will provide the necessary definitions of the research probelm, including HINs and the network embedding.

In the real world, many networks consist of the multiple types of nodes and/or edges, and we call such networks as *Heterogeneous Information Networks (HINs)* [1]. Figure 1.(a) provides a simple heterogeneous movie information network which consists of 6 types nodes (i.e. Movies(M), Users(U), Tags(T), Directors(D), Actors(A) and Countries(C).) and 5 types edges (M-A, M-C, M-D, M-T and M-U.)

Definition 1: ***Heterogeneous Information Networks*** can be modelled as the graph $G = (V, E, X, T_v, T_e)$, where $v_i \in V$, $e_{i,j} \in E$ are the nodes and edges connecting two nodes v_i and v_j in the graph respectively. $X \in R^{n \times f}$ is node feature matrix, where n is the number of nodes and f is the dimension of node features. T_v and T_e are the types of nodes/edges sets respectively.

Definition 2: ***Network Embedding*** aims to map the node $v_i \in V$ in the graph $G = (V, E, X, T_v, T_e)$ into a low dimensional vector $h_i \in R^{1 \times d}$, where d is the dimension of the node embedding vector.

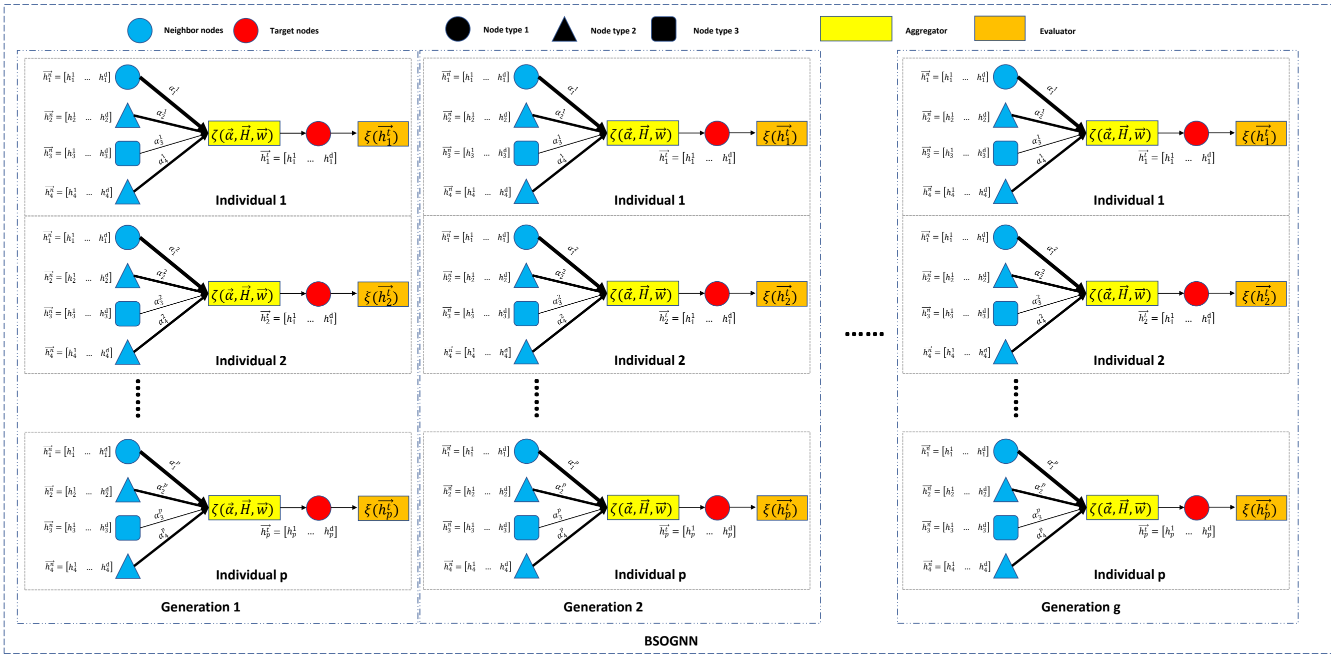

Fig. 2. The architecture of BSOGCN. Blue represents the neighbor nodes, and red represents the target nodes. Nodes with different shapes represent the different types of nodes. $\vec{h^n}$ represents the neighbor nodes' feature vector, $\vec{h^t}$ represents the target nodes' feature vector. $\vec{\alpha}$ represents the neighbor nodes aggregating weights. ζ and ξ represent the aggregator and downstream tasks related evaluation metric (i.e. evaluator) respectively.

IV. BSOGCN

In this section, we will introduce the proposed graph embedding algorithm BSOGCN and the notations used in this paper are summarized in Table I. BSOGCN aims to address the problem that most existing GCNs [17] cannot well discriminate the importance of the different types of nodes in neighbor nodes feature aggregating process for HINs. The key idea of BSOGCN is to assign different neighbor nodes aggregating weights $\vec{\alpha}$ to each neighbor node to discriminate the importance of the different types of nodes, and use BSO [18] algorithm to optimize $\vec{\alpha}$.

Figure 2 provides the architecture of BSOGCN. We will use a single BSOGCN unit (i.e. an arbitrary individual of the population in Figure 2.) to introduce BSOGCN. The input of BSOGCN is a HIN $G = (V, E, X, T_v, T_e)$ and the outputs are node embeddings $H \in R^{n \times d}$, where n is the number of nodes and d is the dimension of node embedding. Specifically, the i-th node embedding vector $\vec{h}_i^k$ at k-th generation can be updated as equation (1):

$$\vec{h}_i^k = ReLU(\zeta(\vec{\alpha}_i^{k-1}, H^{k-1}, W^{k-1})) \tag{1}$$

where $ReLU$ [22] is the nonlinear activation function, ζ is the neighbor node aggregating function (i.e. aggregator), $\vec{\alpha}_i^{k-1}$ is the neighbor node aggregating weights vector of i-th node at $k-1$-th generation, H^{k-1} and W^{k-1} are the node embeddings matrix and the parameters matrix of GCN at $k-1$-th generation respectively, and $H^0 = X$. The aggregator ζ is defined as equation (2):

$$\zeta(\vec{\alpha}_i^{k-1}, H^{k-1}, W^{k-1}) = softmax(A_i \circ \vec{\alpha}_i^{k-1}) \cdot H^{k-1} \cdot W^{k-1} \tag{2}$$

where $softmax$ is the normalization function defined as equation (3) and A_i is the i-th node adjacent vector.

$$softmax(Z) = \frac{exp(z_i)}{\sum_i^Z exp(z_i)} \tag{3}$$

The obtained node embeddings are utilized as the new node feature vectors which are input into the downstream tasks related evaluator ξ (e.g. the $softmax$ function for node multi-class classification task). Thus, the parameters W of GCN can be trained using the gradient descent (e.g. Stochastic Gradient Descent (SGD) [23]) with the cross-entropy loss function $\mathcal{L}$ defined as equation (4).

$$\mathcal{L} = -\sum_{c=1}^{M} y_c log(softmax(\vec{h_c})) \tag{4}$$

where M is the number of trained nodes and y_c is the node label. $\mathcal{L}$ can also be utilized as the fitness function for BSO algorithm to generate the new neighbor node aggregating weight vectors $\vec{\alpha}_i^k$ for i-th node at k-th generation as equation (5):

$$\vec{\alpha}_i^k = \vec{\alpha}_i^{k-1} + rand() \tag{5}$$

where $rand()$ is a random function to generate uniformly distributed random numbers in the range $[0, 1)$. The procedure of BSOGCN is given in Algorithm 1.

TABLE I
THE NOTATIONS USED IN THIS PAPER.

Notations	Descriptions
G	The heterogeneous information network
V	The nodes set
$v_i \in V$	The i-th node in nodes set V
E	The edges set
e_{ij}	The edge connecting i-th node and j-th node
$X \in R^{n \times f}$	The node features matrix
$\vec{x_i} \in R^{1 \times f}$	The i-th node features vector
T_v	The types of nodes set
T_e	The types of edges set
n	The number of nodes
f	The dimension of node features
$H \in R^{n \times d}$	The node embeddings matrix
$\vec{h_i} \in R^{1 \times d}$	The i-th node embedding vector
d	The dimension of node embeddings
p	The population size
$A \in R^{n \times n}$	The adjacent matrix of G
$A_i \in R^{1 \times n}$	The i-th node adjacent vector
$\vec{h^n} \in R^{1 \times n}$	The neighbor node embedding vector
$\vec{h^t} \in R^{1 \times n}$	The target node embedding vector
$\vec{\alpha} \in R^{1 \times n}$	The neighbor node aggregating weights vector
ζ	aggregator
ξ	The downstream task related evaluation metric
$N(v_i)$	The neighbor nodes set of i-th node
$\circ$	The hadamard product
W	The parameters of GCN

Algorithm 1 The procedure of BSOGCN

Input: :
A HIN $G = (V, E, X, T_v, T_e)$
Output: :
Node embedding matrix $H \in R^{n \times d}$

```
H^0 = X;
Randomly initialize the node aggregating weights α⃗ for all nodes;
Randomly initialize the parameters W of GCN;
for k=1...K do
    for v_i ∈ V do
        h⃗_i^k ← ReLU(ζ(α⃗_i^{k-1}, H^{k-1}, W^{k-1}))
    end for
    L^k ←-∑_{c=1}^{M} y_c log(softmax(h⃗_c))
    W^k ←SGD(L^k, W^{k-1})
    α⃗^k ←BSO(L^k, α⃗^{k-1}))
end for
```

V. EXPERIMENTS

For validating the performance of BSOGCN, we conduct experiments on three real-world HINs datasets under various scenarios on node multi-class classification task with both inductive and transductive setting. The rest of this section will introduce the more detailed information about the datasets, the comparison algorithms, the experimental setting, and the results.

A. Dataset

The statistic information of the HINs datasets has been summarized in Table II.

TABLE II
THE STATISTIC INFORMATION OF THE HINS DATASETS

Datasets	Nodes	Edges	Label
DBLP	Author(A): 10454 Paper(P): 6980 Venue(V): 75	A-A: 12152 A-P: 14150 A-V: 12940 P-P: 845 P-V: 6980	7
YELP	User(U): 20000 Business(B): 16216 Category(C): 460 City(T): 10 Compliment(P): 11	U-U: 519960 U-P: 107297 U-B: 538274 B-T: 16216 B-C: 53206	10
Hetrec2011-moveilens (HM)	Movie(M): 10197 User(U): 2113 Tag(T): 5297 Director(D): 4060 Actor(A): 10000 Country(C): 72	M-A: 103372 M-C: 10197 M-D: 10155 M-T: 51795 M-U: 855598	19

- **DBLP Dataset**: DBLP [24] is a bibliographic information network dataset which consists of the three types of nodes (i.e. the Author (A) node, the Paper (P) node and the Venue (V) node) and the five types of edges (i.e. A-A, A-P, A-V, P-P and P-V). We extract the papers published from 2009 to 2013 year in 7 research fields on 13 conferences[1]. For each paper node, we use Word2vec [25] to transform its abstract into the 300 dimensional feature vector. The features of other types of nodes are initialized randomly with the same dimension as the paper nodes.
- **YELP Dataset**: YELP[2] is a business review information dataset which consists of the five types of nodes (i.e. the User (U) node, the Business (B) node, the Category (A) node, the City (C) node and the Compliment (P) node) and the five types of edges (i.e. U-U, U-P, U-B, B-C and B-A). We select the business nodes from the top 10 cities with the most restaurants and use their cuisines[3] as their labels. The features of the business nodes are the bag-of-words vectors with 1000 words extracted from the business name. The features of other types of nodes are initialized randomly with the same dimension as the business nodes.
- **HM Dataset**: HM[4] is a movie information network dataset based on MovieLens, IMDb[5] and Rotten Toma-

[1] 13 conferences are ”ICDE”,”SIGMOD”,”KDD”,”ICDM”, ”IJCAI”,”ICML”,”CVPR”,”ICCV”,”SODA”,”ACL”,”EMNLP”, ”COLING”,”SOSP”, which belong to 7 research fields including DB,DM,AI,NLP,CV,OS,CT

[2] https://www.yelp.com/dataset/challenge

[3] We select the restaurants with only one cuisine, and the cuisines include ”American”, ”Greek”, ”Mexican”, ”Italian”, ”Chinese”, ”Japanese”, ”Thai”, ”Indian”, ”Canadian”, ”Meddle Eastern”.

[4] http://www.grouplens.org

[5] http://www.imdb.com

TABLE III
THE TRANSDUCTIVE NODES MULTI-CLASS CLASSIFICATION RESULTS ON YELP DATASET

Yelp	**Algorithms**	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
Micro-f1	LR	0.1318	0.1399	0.1473	0.1468	0.1508	0.1505	0.1680	0.1646	0.1504
	GCN	0.0999	0.1268	0.1521	0.0988	0.1407	0.2355	0.2483	0.2272	0.2577
	GAT	0.5035	0.5060	0.5041	0.4654	0.5054	0.5067	0.5067	0.5059	0.4988
	BSOGCN	**0.5070**	**0.5075**	**0.5082**	**0.5091**	**0.5091**	**0.5084**	**0.5079**	**0.5065**	**0.5043**
Macro-f1	LR	0.0108	0.0161	0.0110	0.0139	0.0173	0.0165	0.0137	0.0187	0.0153
	GCN	0.0476	0.0572	0.0576	0.0464	0.0576	**0.0736**	**0.0719**	0.0663	0.0743
	GAT	0.0673	0.0673	0.0674	0.0675	0.0675	0.0674	0.0674	**0.0672**	**0.0745**
	BSOGCN	**0.0681**	**0.0676**	**0.0683**	**0.0730**	**0.0685**	0.0674	0.0676	**0.0672**	0.0685

TABLE IV
THE TRANSDUCTIVE NODES MULTI-CLASS CLASSIFICATION RESULTS ON HETREC2011-MOVIELENS DATASET

HM	**Algorithms**	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
Micro-f1	LR	0.3449	0.3570	0.3723	0.3708	0.3798	0.3764	0.3775	0.3804	0.3919
	GCN	0.2603	0.4214	0.2996	0.4856	0.4771	0.4670	0.4740	0.4764	0.4812
	GAT	0.2274	0.2147	0.2215	0.2392	0.3317	0.4105	**0.4677**	**0.4912**	**0.4892**
	BSOGCN	**0.4683**	**0.4981**	**0.4977**	**0.4933**	**0.4981**	**0.4966**	0.4659	0.4832	**0.4892**
Macro-f1	LR	0.0112	0.0164	0.0107	0.0118	0.0160	0.0174	0.0197	0.0106	0.0140
	GCN	0.0375	**0.0469**	**0.0424**	**0.0429**	0.0412	0.0486	0.0435	0.0479	0.0482
	GAT	0.0249	0.0237	0.0270	0.0319	**0.0440**	**0.0504**	0.0472	0.0475	0.0472
	BSOGCN	**0.0414**	0.0395	0.0404	0.0409	0.0401	0.0438	**0.0483**	**0.0512**	**0.0506**

TABLE V
THE INDUCTIVE NODES MULTI-CLASS CLASSIFICATION RESULTS ON DBLP DATASET.

DBLP	**Algorithms**	**1@4**	**2@3**	**3@2**	**4@1**
Micro-f1	LR	0.1321	0.1317	0.1344	0.1402
	GCN	**0.4157**	0.3557	0.3996	0.3944
	GAT	0.1459	0.1423	0.1331	0.1371
	BSOGCN	0.3710	**0.3692**	**0.4250**	**0.4309**
Macro-f1	LR	0.0245	0.0256	0.0211	0.0298
	GCN	0.0954	**0.1042**	**0.1040**	**0.1121**
	GAT	0.0808	0.0797	0.0789	0.0819
	BSOGCN	**0.1051**	0.0917	0.0935	0.0986

toes movie review system[6]. It consists of six types of nodes (i.e. the Movie (M) node, the User(U) node, the Tag(T) node, the Director(D) node, the Actor(A) node and the Country(C) node) and the five types of edges (i.e. M-A, M-C, M-D, M-T and M-U). The movies with only one genre(label) are extracted as training/testing nodes. The features of the movie nodes are the bag-of-words vectors with 1000 words extracted from the movies' title. The features of other types of nodes are initialized randomly with the same dimension as the movie nodes.

B. Comparison Algorithms

- **Logistic Regression (LR)**: LR the baseline algorithm which directly use the original feature vectors of the nodes to train and test.
- **GCN**: GCN [3] is the original graph convolutional based method which assigns the same aggregating weights to the neighbor nodes, which cannot discriminate the importance of different types of nodes in HINs.
- **GAT**: GAT [19] has the similar idea with the proposed method, and it utilizes the attention mechanism to discriminates the importance of different types of nodes. However, it utilizes the gradient based optimization method to optimize the aggregating weights.

C. Experimental Setting

The performance of the proposed method BSOGCN's is evaluated on the three nodes multi-class classification tasks for both inductive and transductive setting. All the experiments have been repeated 10 times and use average Micro-f1 and Macro-f1 scores as the evaluation metric.

The Yelp and HM datasets are utilized for the transductive node multi-class classification tasks. The business nodes and the movie nodes are utilized as the target nodes for classification prediction respectively. We vary the size of the training nodes set from 0.1 to 0.9, and the rest are utilized as the testing nodes set. The experimental results are show in Table III and IV respectively.

The DBLP dataset is utilized for the inductive node multi-class classification task. The paper nodes are utilized as the target nodes for classification prediction. We vary the size of

[6] http://www.rottentomatoes.com

the training nodes set from 2009 to 2012 years, and the rest of years are utilized as the testing nodes set, which is denoted by $TrainingYear@TestingYear$. The experimental results are shown in Table V.

The hyperparameters setting of BSOGCN are summarized as Tabel VI.

TABLE VI
THE HYPERPARAMETERS SETTING OF BSOGCN.

Hyperameters	**Descriptions**	**Values**
d	The dimension of node embeddings	128
p	The population size	100
K	The epochs/generations of GCNBSO	200
L	The layer of GCN	1
$p_{Generation}$	The pre-determined probability of BSO	0.2
$p_{OneCluster}$	The pre-determined probability of BSO	0.8
$p_{TwoCluster}$	The pre-determined probability of BSO	0.2
c	The number of clusters of BSO	5

D. Results

As shown in Table III, Table IV and Table V, We can clearly observe that: (1) The proposed method BSOGCN can obtain the competitive performance against the other state-of-the-art methods on both inductive and transductive node multi-class classification tasks. (2) The size of training data has a minor effect on BSOGCN, which shown that BSOGCN can obtain a promising results with few labeled data. (3) Comparing with the results of GCN, we can found that the different types of neighbor nodes have the various effect on the target nodes on HINs. (4) Comparing with the results of GAT, we can found that the BSO is a alternative algorithm to optimize the neighbor nodes aggregating weights.

VI. CONCLUSION

In this paper, to address the problem that the different types of nodes have different impact on the target nodes for HINs embedding, we propose a novel HINs embedding method BSOGCN. It assigned the different aggregating weights to the different types of neighbor nodes, and utilized the BSO algorithm to optimize the weights. The experimental results showed that BSOGCN can obtain the competitive results against other state-0f-the-art methods.

In the future, we plan to explore how the BSOGCN can be applied to the large-scale dynamic HINs and we also intend to explore the reasons that BSOGCN can obtain the promising results.

VII. ACKNOWLEDGMENT

This work is partially supported by National Key RD Program of China under the Grant No. 2017YFC0804003, National Science Foundation of China under grant number 61761136008, Shenzhen Peacock Plan under Grant No. KQTD2016112514355531, Program for Guangdong Introducing Innovative and Entrepreneurial Teams under grant number 2017ZT07X386, and the Science and Technology Innovation Committee Foundation of Shenzhen under the Grant No.ZDSYS201703031748284.

REFERENCES

[1] Y. Sun and J. Han, *Mining Heterogeneous Information Networks: Principles and Methodologies*. Morgan Claypool, 2012. [Online]. Available: https://ieeexplore.ieee.org/document/6812698

[2] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "PathSim: Meta Path-Based Top-K Similarity Search in Heterogeneous Information Networks," p. 12, 2011.

[3] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," *arXiv:1609.02907 [cs, stat]*, Sep. 2016, arXiv: 1609.02907. [Online]. Available: http://arxiv.org/abs/1609.02907

[4] F. Tian, B. Gao, Q. Cui, E. Chen, and T.-Y. Liu, "Learning Deep Representations for Graph Clustering," p. 7, 2014.

[5] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale Information Network Embedding," *Proceedings of the 24th International Conference on World Wide Web - WWW '15*, pp. 1067–1077, 2015, arXiv: 1503.03578. [Online]. Available: http://arxiv.org/abs/1503.03578

[6] C. Zhou, Y. Liu, X. Liu, Z. Liu, and J. Gao, "Scalable Graph Embedding for Asymmetric Proximity," p. 7, 2017.

[7] A. Mahmood, M. Small, S. A. Al-Maadeed, and N. Rajpoot, "Using Geodesic Space Density Gradients for Network Community Detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 4, pp. 921–935, Apr. 2017.

[8] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation Learning on Graphs: Methods and Applications," *arXiv:1709.05584 [cs]*, Sep. 2017, arXiv: 1709.05584. [Online]. Available: http://arxiv.org/abs/1709.05584

[9] S. Wold, K. Esbensen, and P. Geladi, "Principal Component Analysis," p. 16.

[10] I. Borg and P. Groenen, "Modern multidimensional scaling: Theory and applications," *Journal of Educational Measurement*, vol. 40, no. 3, pp. 277–280, 2003.

[11] J. B. Tenenbaum, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.

[12] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[13] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013. [Online]. Available: http://arxiv.org/abs/1301.3781

[14] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online Learning of Social Representations," *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14*, pp. 701–710, 2014, arXiv: 1403.6652. [Online]. Available: http://arxiv.org/abs/1403.6652

[15] A. Grover and J. Leskovec, "node2vec: Scalable Feature Learning for Networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '16*. San Francisco, California, USA: ACM Press, 2016, pp. 855–864. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2939672.2939754

[16] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable Representation Learning for Heterogeneous Networks," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '17*. Halifax, NS, Canada: ACM Press, 2017, pp. 135–144. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3097983.3098036

[17] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun, "Graph Neural Networks: A Review of Methods and Applications," *arXiv:1812.08434 [cs, stat]*, Dec. 2018, arXiv: 1812.08434. [Online]. Available: http://arxiv.org/abs/1812.08434

[18] Y. Shi, "Brain storm optimization algorithm," in *Advances in Swarm Intelligence*, Y. Tan, Y. Shi, Y. Chai, and G. Wang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 303–309.

[19] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph Attention Networks," *arXiv:1710.10903 [cs, stat]*, Oct. 2017, arXiv: 1710.10903.

[20] T.-y. Fu, W.-C. Lee, and Z. Lei, "HIN2vec: Explore Meta-paths in Heterogeneous Information Networks for Representation Learning," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management - CIKM '17*. Singapore, Singapore: ACM Press, 2017, pp. 1797–1806. [Online]. Available: http://dl.acm.org/citation.cfm?doid=3132847.3132953

[21] Z. Zhang, P. Cui, and W. Zhu, "Deep Learning on Graphs: A Survey," *arXiv:1812.04202 [cs, stat]*, Dec. 2018, arXiv: 1812.04202. [Online]. Available: http://arxiv.org/abs/1812.04202

[22] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *arXiv preprint arXiv:1505.00853*, 2015.

[23] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[24] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: Extraction and mining of academic social networks," in *KDD'08*, 2008, pp. 990–998.

[25] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50, http://is.muni.cz/publication/884893/en.