EasyChair Preprint

№ 3331

Deep Network Embedding Method Based on Community Optimization

Yafang Li, Ye Liang, Weiwei Feng, Baokai Zu and Yujian Kang

May 4, 2020

基于社区优化的深度网络嵌入方法

李亚芳[1] 梁烨[1] 冯韦玮[1] 祖宝开[1] 康玉健[1]

[1]（北京工业大学信息学部 北京 100123）

通讯地址： yafangli@bjut.edu.cn

Deep Network Embedding Method Based on Community Optimization

Yafang Li[1], Ye Liang[1],Weiwei Feng[1],Baokai Zu[1],Yujian Kang[1]

[1]（*Faculty of Information Technology, Beijing University of Technology, Beijing*100123）

Abstract With the rapid development of modern network communication and social media technology, the networked big data is difficult to apply due to the lack of efficient and available node representation. Network representation learning is widely concerned by transforming high-dimensional sparse network data into low-dimensional, compact and easy to use expressions. However, the existing methods get the low-dimensional feature vector of nodes, and then use it as the input of other applications (classification, clustering, prediction, visualization, etc.) for further analysis, which is lack of specific application in designing model. In this paper, a deep auto-encoder clustering model, CADNE, was proposed to represent the low dimensional features of nodes based on community structure optimization. This method can learn the low-dimensional representation of nodes and the indicator vector of their communities at the same time, so that the low-dimensional representation of nodes can not only maintain the neighborhood characteristics of the original network structure, but also maintain the clustering characteristics of nodes. Experiments on multiple data sets show that the CADNE method has better ability of low dimensional representation of nodes.

Keywords large-scale complex networks; community structure;deep learning; network embedding

摘要 随着现代网络通信和社会媒体等技术飞速发展，网络化的大数据由于缺少高效可用的节点表示，难以应用。网络表示学习通过将高维稀疏难于应用的网络数据，转化为低维、紧凑、易于应用的表达而受到广泛关注。但已有方法得到节点低维特征向量后，再将其作为其他应用（分类、聚类、预测、可视化等）的输入进一步分析，缺少针对具体应用设计模型。本文，针对网络社区发现，提出基于社区结构优化进行节点低维特征表示的深度自编码聚类模型 CADNE。该方法能够同时学习节点的低维表示和节点所属社区的指示向量，使得节点的低维表示不仅能够保持原始网络结构中的近邻特性，而且能够保持节点聚类特性。与已有经典网络表示学习方法在多个数据集上的实验表明：CADNE 方法具有较好的节点低维表示能力。

关键词 大规模复杂网络；社区结构；深度学习；网络表示学习

中图法分类号 TP391

网络已成为最常见的信息载体和表示形式，在我们的日常生活与工作中无处不在，如社交网络、论文合作者网络、通信网络以及生物网络等。这些网络不仅包含复杂的链接关系，而且附着各类描述数据对象的内容属性，形成复杂的属性网络。以社交网络为例，除了用户之间的关注或朋友关系建立的链接关系外，

每个用户还包含年龄、性别、爱好、职业等内容信息。在社交网络已成为人工智能应用焦点的大背景下，如何同时利用拓扑结构和内容属性，对网络数据进行研究与分析，已经受到社会各界的广泛关注，在用户画像、内容推荐、舆情监测、生物分子功能团识别等众多领域具有潜在应用价值[1]。

然而，随着互联网技术以及大数据的蓬勃发展，以微博、微信、Twitter 和 Facebook 为代表的社交网络进入亿级节点时代，不仅网络规模不断扩大、链接关系更加复杂、存在缺失数据和噪声信息，内容属性也呈现高维、稀疏特性，这为大规模网络的相关研究又提出了更新、更大的挑战。对于这类大规模稀疏属性网络，传统将网络结构表示为高维稀疏邻接矩阵，进行网络分析的方法显得力不从心[2]：计算复杂度高；节点通过边关联，难以并行化；难以直接应用机器学习方法，对于大规模网络的高维、稀疏向量，已有的统计学习方法会花费更多的运行时间和计算空间，使得许多先进的研究成果无法直接应用到现实的网络环境中。

网络嵌入（网络表征学习）方法是解决传统方法缺陷的有效方式，在保留结构信息的前提下，为网络中的每个节点学习一个低维、稠密的连续特征向量表示。通过将网络数据表示成一种高效合理的形式，不仅有助于更好理解节点之间的语义关联，而且能够有效解决大规模网络中的稀疏性，也可作为经典机器学习模型的输入，采用已经成熟的模型和方法将其运用于后续节点分类、节点聚类、链接预测以及可视化等网络分析任务中，对解决现实网络中的实际应用问题具有重要意义，如通过节点分类构建用户推荐系统；通过社区发现进行舆情监测；通过链路预测推测蛋白质之间可能存在的相互作用关系以推动疾病的治疗。

1 相关工作

近来，针对网络结构特征的网络表示学习算法相继提出，大致可分为：基于因子分解的方法[4-11]、基于神经网络的表示方法[12-19]。基于因子分解的方法首先构造关系矩阵，通常为邻接矩阵、Laplacian 矩阵、节点转移概率矩阵或其他相似度矩阵，通过对关系矩阵的分解得到节点的低维向量表示。该类方法可进一步分为：1）特征值分解方法（特征向量表示方法），如局部线性表示 LLE[4]、LE [5]、SPE [6]；2）矩阵分解方法。主要包括图分解方法 GF[7]、GraRep[8]、HOPE[9]、M-NMF[10]、NEU[11]。GraRep 通过构建 k 阶相似度矩阵，往往得到较好的效果，但算法的计算复杂度更高。NEU 则采用相似性方法提高基于高阶相似度矩阵进行节点表示的效率。M-NMF 通过融合模块度的非负矩阵分解方法，将社团结构信息纳入网络表示学习中。基于因子分解的方法，构建的关系矩阵包括高阶节点链接信息时，能够显著提升节点表示的效果，但计算和存储效率相对较低，难以扩展到大规模网络。而且基于因子分解方法只关注节点间线性结构关系，往往是不够的，网络的形成非常复杂的过程，节点间常具有非线性复杂结构关系。因此，网络研究者利用神经网络建模节点表示之间的非线性关系。

DeepWalk[12]是第一个采用神经网络进行网络表示学习的方法，通过随机游走得到网络结构的线性序列，进一步采用训练词向量的神经网络模型 SkipGram 方法进行网络中节点的表示学习。在 DeepWalk 的基础上，相继提出了 node2vec[13]、HARP[14]、DDRW[15]、NEES[16]方法。DeepWalk 及其扩展方法通过某种随机游走策略自动地抽样网络中节点的路径，然后通过神经网络模型得到节点的表示，但该类方法属于浅层神经网络方法，难以充分捕捉现实世界复杂网络中节点间的高度非线性关系。进而，Wang 等人提出基于深度自编码节点表示方法 SDNE[17]，通过综合考虑网络拓扑结构的一阶和二阶相似度，取得较好的节点表示性能。DNGR[18]构建节点间 PPMI 关联矩阵，通过深层降噪自编码模型学习节点的低维向量表示。基于深度学习的网络表示方法更具有更强的节点表示能力，不仅能够学习节点间复杂的非线性关系，而且可通过高效优化方法对模型参数进行求解。

通过节点低维特征表示的学习，为大规模网络的分析和处理提供一条可行解决思路。但已有方法得到的节点低维特征向量后，需要将其作为其他应用（分类、聚类、预测、可视化等）的输入来进一步分析，采用的是两步走策略；缺少针对具体应用来设计模型，因为不同的应用场景对学习属性的选择通常有不同的要求。因此，本文针对网络节点聚类（社区发现）的应用，基于深度神经网络的自动编码器模型SDNE，结合网络数据的局部和全局拓扑结构特性以及深度嵌入聚类算法DEC(deep embedding clustering)[19]，提出节点低维特征表示和社区结构优化的深度网络嵌入模型 CADNE(Community-AwareDeep Network Embedding)。该方法能够同时学习节点的低维表示向量和节点所属社区的指示向量，使得节点的低维表示不仅能够保持原始网络结构中的近邻特性，而且能够保持原始拓扑空间的社区结构。

2 社区优化的深度网络嵌入 CADNE

2.1 基本定义

定义 1. 网络（Network）：网络可描述为$G=(V,E)$，$V=\{v_1,v_2,\dots,v_n\}$表示网络中n个节点组成的集合，E表示节点间边的集合。每条边$e\in E$是一对包含权重A_{ij}的有序节点对$e=(u,v)$，如果节点v_i和v_j间不存在链接，则$A_{ij}=0$；若存在链接，对于无权网络$A_{ij}=1$，对于有权网络为$A_{ij}>0$。

定义 2.一阶相似性（first-order proximity）：描述任意两个节点之间的局部结构相似度，如果两个节点间存在链接，其一阶相似性度$A_{ij}>0$，否则为0。

一阶近邻描述网络中存在直接链接的节点对之间的相似度，只关注两个节点对之间是否存在直接的链接。然而，现实网络的链接关系往往非常稀疏，而且存在很多非常相似的节点对但其间没有直接链接关系，因此，引入二阶相似性作为补充以描述全局网络结构的相似性。

定义 3.二阶相似性（second-order proximity）：描述任意两个节点与网络中其他节点链接结构的相似性，对于节点v_i和v_j，$A_i=\{A_{i1},A_{i2},\dots,A_{in}$和$A_j=\{A_{j1},A_{j2},\dots,A_{jn}\}$分别表示两个节点与网络中其他节点的一阶相似性，则节点v_i和v_j的二阶相似性为A_i和A_j的相似度。

可见，如果两个节点共有的邻居节点越多，其二阶相似性越大，这两个节点越相似；若两个节点间不存在共同的邻居链接，其二阶相似度为0。通过二阶相似性可度量网络中未存在直接链接关系的节点对之间的相似度，度量网络节点的全局结构相似度。本文同时考虑节点间的一阶相似性和二阶相似性，对网络中的节点进行映射表示。

定义 4.网络嵌入（network embedding）：对于网络$G=(V,E)$，学习映射函数$f:V\to R^d$将每个节点映射为d维$(d<n)$特征空间的表示向量。

2.2CADNE 模型框架

给定无向网络$G=(V,E)$，节点间的链接关系可用邻接矩阵表示$A=\{A_1,A_2,\dots,A_n\}$描述，可见，原始网络拓扑结构空间中，每个节点通过n维向量进行表示。本文，我们提出基于深度嵌入聚类进行社区优化的深度网络表示模型，学习节点在d维低维特征空间的表示，同时得到节点的社区划分，整个模型框架如图1。

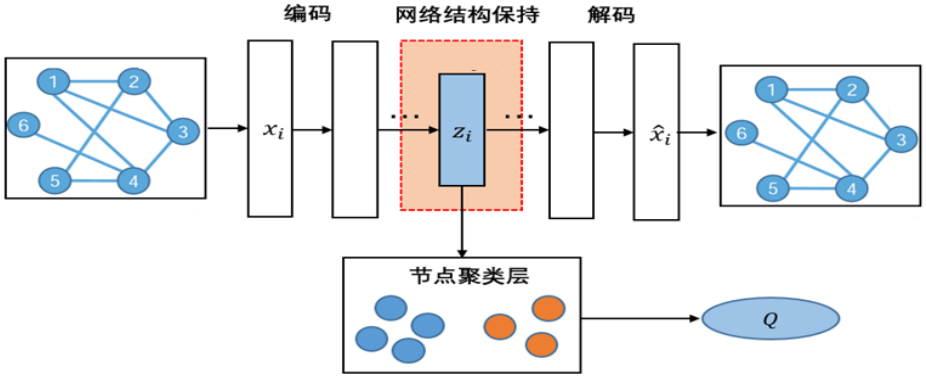


Fig.1 The framework of CADEN model

图 1CADNE 模型框架

该模型主要由两部分组成，第一部分是深度自编码模型通过非线性激活函数进行参数训练，将节点映射为易于计算的低维、稠密向量表示，以保持原始网络结构中节点间高度非线性关系，在映射过程中，保持网络节点一阶相似性（局部）及二阶相似性（全局）的拓扑特性；第二部分是基于 DEC 模型，利用节点聚类结构对节点低维表示进一步优化，使得节点低维表示过程中仍保持节点聚集特性，通过交替迭代更新深度自编码模型的编码过程以及节点聚类，得到社区结构优化的节点低维表示。

2.3 保持网络拓扑结构

为了使低维表示后的节点在新的特征空间中，仍保持原网络拓扑结构中的局部近邻特性，综合考虑节点间的一阶相似性以及二阶相似性，采用深度自动编码实现节点稀疏表示的降维，具体从以下两个方面构建模型损失函数。

设x_i表示根据网络结构得到的模型输入，如果输入为邻接矩阵，则$A_{\mathrm{i}} = x_i$，表示节点v_i与网络中其它节点的链接关系向量，及节点的全局链接结构特征，通过将邻居矩阵作为输入，节点低维表示以及网络重建过程中，使得节点在原始拓扑结构中具有相似链接特征的节点的低维表示也尽可能相似。假设深度自动编码网络有K层，则每层的隐含表示为：

$$z_i^{(1)} = \sigma(W^{(1)}x_i + b^{(1)})$$

$$z_i^{(k)} = \sigma(W^{(k)}z_i^{(k-1)} + b^{(k)}), k = 1,2,\dots,K$$

得到最深层的z_i为节点的低维向量表示，通过逆向解码得到自动编码网络的输出$\hat{x}_i$：

$$\hat{z}_i^{(k-1)} = f(M^{(k-1)}\hat{z}_i^{(k)} + d^{(k-1)}), k = 1,2,\dots,K$$

$$\hat{x}_i = \sigma(M^{(1)}\hat{z}_i^{(2)} + d^{(1)})$$

其中，$\sigma(x)$、$\mathrm{f}(x)$为非线性的激活函数，$\theta_{enc} = \{\mathrm{W},\mathrm{b}\}$，$\theta_{dec} = \{M, d\}$是待学习的神经网络参数。目标是根据新的节点低维表示z_i，最小化输入x_i和输出$\hat{x}_i$的重构误差，得到网络二阶相似性的目标函数：

$$\min_{\theta\in\Theta}\sum_{i=1}^{n}\|\hat{x}_i - x_i\|_2^2$$

然而，现实网络中链接非常稀疏，只有极少量的边被观测到，因此邻接矩阵中零元素个数远多于非零元素的数目。如果直接使用邻接矩阵作为模型的输入，过多的零元素将会影响原始网络的低维表示以及重建过程，通过最小化重构误差会使得节点的重建表示$\hat{X}$倾向于重建很多零元素。因此，在网络低维表示和重建过程中，我们重点关注邻接矩阵中的非零元素，定义二阶相似性目标损失函数L_{2nd}为：

$$\min_{\theta\in\Theta}\sum_{i=1}^{n}\|(\hat{x}_i - x_i)\odot b_i\|_2^2 = \left\|(\hat{X} - X)\odot b_i\right\|_2^2$$

其中，$\odot$是哈达玛积，$b_i = \{b_{ij}\}_{j=1}^{n}$，如果邻接矩阵元素$a_{ij} = 0$，$b_{ij} = 1$，否则$b_{ij} = \beta > 1$。通过该二阶相似性的目标约束，使得原始网络拓扑空间中具有相似全局链接结构关系的节点的低维表示也尽可能相似。

为了保持原始网络空间节点的局部结构，节点低维表示映射过程中，使得存在直接链接的节点对的低维表示尽可能相似，因此对这类节点对进行约束，如果其低维表示的距离较远则引入较大的惩罚。我们构建一阶相似性损失函数L_{1st}，定义$d_{ii} = \sum_{j=1}^{n}\boldsymbol{X}_{ij}$,$L = D^{-\frac{1}{2}}(D - X)D^{-\frac{1}{2}}$，定义优化目标为：

$$\min_{\theta\in\Theta}\frac{1}{2}\sum_{i,j=1}^{n}A_{ij}\left\|\frac{z_i}{\sqrt{d_{ii}}} - \frac{z_j}{\sqrt{d_{jj}}}\right\|_2^2 = \min_{\theta\in\Theta}Tr(Z^T LZ)$$

为了使网络中节点映射为低维特征空间表示过程中，同时保持网络中局部以及全局结构，将一阶相似性与二阶相似性综合得到目标函数为：

$$L_{ae} = L_{1st} + \gamma L_{2nd}$$

2.4 保持网络潜在聚类结构

在低维表示空间引入聚类损失，使学到的网络嵌入能够更好地保持网络聚类结构，基于深度聚类算法 DEC，将节点聚类融合到节点低维表示模型，利用节点聚类结构对低维表示进行进一步优化。将低维表示的节点向量$z_i(i = 1,2\dots,n)$进行聚类，设节点z_i属于类u_j的概率为$q_{ij}(q_{ij}\in Q)$，表示节点$\mathrm{z_i}$属于类中心u_j的相似度，学生 *t*-分布（Student's t-distribution）为：

$$q_{ij} = \frac{(1 + \left|\left|z_i - u_j\right|\right|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1 + \left|\left|z_i - u_{j'}\right|\right|^2/\alpha)^{-\frac{\alpha+1}{2}}}$$

其中$\alpha = 1$，定义$f_j = \sum_i q_{ij}$，引入辅助目标分布P：

$$p_{ij}=\frac{q_{ij}^{s}/f_j}{\sum_{j'}q_{ij'}^{s}/f_{j'}}$$

因此，将节点低维表示的类分布Q与目标分布P拟合，采用 KL 散度衡量，得到目标函数：

$$L=KL[P||Q]=\sum_i\sum_j p_{ij}log\frac{p_{ij}}{q_{ij}}$$

通过随机梯度下降（SGD）交替更新聚类中心u以及编码层模型参数θ_{enc}，对模型进行细致优化，通过对每个节点低维表示的梯度$\frac{\partial L}{\partial z_i}$通过后向传播，得到对编码器参数梯度$\frac{\partial L}{\partial \theta_{enc}}$，实现编码器参数的更新，其中对每个节点低维表示$z_i$以及每个类聚类中心$u_j$的梯度为：

$$\frac{\partial L}{\partial z_i}=\frac{\alpha+1}{\alpha}\sum_{\mathrm{j}}(1+\frac{z_i-\mathrm{u_j}}{\alpha})^{-1}\times(p_{ij}-q_{ij})(z_i-\mathrm{u_j})$$

$$\frac{\partial L}{\partial u_i}=-\frac{\alpha+1}{\alpha}\sum_{i}(1+\frac{z_i-u_j}{\alpha})^{-1}\times(p_{ij}-q_{ij})(z_i-u_j)$$

2.5CADNE 算法实现

模型的训练主要分成两部分，在网络拓扑结构保持部分，即模型预训练部分，通过对深度自编码模型的编码器 encoder 以及解码器 decoder 进行训练，采用 Adam 优化目标函数L_{ae}，使得节点低维表示过程中同时保持网络结构的局部以及全局结构特性；第二部分根据节点聚类结构对节点低维表示进行优化，对编码器的编码过程进一步训练，使得节点的低维表示过程保持聚类结构。本文提出的 CADNE 算法流程，如下图所示：

输入：网络$G=(V,E)$的邻接矩阵 A

输出：网络节点低维表示Z，模型参数θ_{enc}、θ_{dec}

初始化编码器 encoder(Enc)参数θ_{enc}，解码器 decoder(Dec)参数θ_{dec}

repeat:

#模型预训练

1：随机小批量（mini-batch）获得模型输入X

2：$Z\leftarrow\mathrm{Enc}(X)$

3：$\mathrm{L_{1st}}=\frac{1}{2}\sum_{i,j=1}^{n}A_{ij}\left\|\frac{z_i}{\sqrt{d_{ii}}}-\frac{z_j}{\sqrt{d_{jj}}}\right\|_2^2=Tr(Z^TLZ)$

4：$\hat{X}\leftarrow\mathrm{Dec}(Z)$

5：$\mathrm{L_{2nd}}=\sum_{i=1}^{n}\|(\hat{\boldsymbol{x}}_i-\boldsymbol{x}_i)\odot\boldsymbol{b}_i\|_2^2=\|(\hat{\boldsymbol{X}}-\boldsymbol{X})\odot\boldsymbol{b}_i\|_2^2$

6：$\mathrm{L_{ae}}=\mathrm{L_{1st}}+\gamma\mathrm{L_{2nd}}$

7：$\theta_{enc}\leftarrow\theta_{enc}-\rho\nabla_{\theta_{enc}}\mathrm{L_{ae}}$

8：$\theta_{dec}\leftarrow\theta_{dec}-\rho\nabla_{\theta_{dec}}\mathrm{L_{ae}}$

#结合嵌入聚类联合训练模型参数

1：$Z\leftarrow\mathrm{Enc}(A)$

2：$u\leftarrow k-means(Z)$

3：$q_{ij}=\frac{(1+\left\|z_i-u_j\right\|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{j'}(1+\left\|z_i-u_{j'}\right\|^2/\alpha)^{-\frac{\alpha+1}{2}}}$

4：$p_{ij}=\frac{q_{ij}^{s}/f_j}{\sum_{j'}q_{ij'}^{s}/f_{j'}}$

5：$L=KL[P||Q]=\sum_i\sum_j p_{ij}log\frac{p_{ij}}{q_{ij}}$

6：$\theta_{enc}\leftarrow\theta_{enc}-\delta\nabla\theta_{enc}L$

7：$u\leftarrow u-\delta\nabla uL$

until converge

3 实验设计与结果分析

为了验证本文提出的基于社区优化的深度网络嵌入方法（CADNE）的有效性, 本节与几个经典的网络表示学习模型进行对比。

3.1 实验设置

本文在数据集上 20-NEWSGROUP[20]、Cora[21]、Citeseer[22]、BlogCatalog[23]上进行测评。各个数据集的相关统计信息说明如表 1 所示。

Table 1 Datasets introduction

表 1 数据集介绍

Datasets	Node	Edge	Class
20-NewsGroup	1727	full-connection	3
Cora	2708	5278	7
Citeseer	3312	4536	6
BlogCatalog	5196	171743	6

为了更好地评价本文所提出的模型方法,在实验部分与六个代表算法进行对比分析，包括：

1） DeepWalk：该方法通过在图中进行随机游走得到的节点序列，将序列输入使用 Skip-Gram 模型得到每个节点的嵌入表示；

2） LINE：该方法通过优化保持一阶相似度和二阶相似度的目标函数来学习每个节点的低维表示向量；

3） SDNE：该方法通过构建深层自编码器保留网络一阶相似度和二阶相似度,学习节点的低维表示；

4） DNGR：构建节点间 PPMI 关联矩阵，通过降噪自编码得到节点的低维向量表示；

5） M-NMF：基于矩阵分解学习节点低维嵌入表示，模型训练过程中考虑了节点的社区结构；

6） DEC：深度嵌入聚类算法，将网络邻接矩阵作为模型输入进行训练，没有考虑网络的拓扑结构信息。

参数设定：为保证算法的对比公平，对比方法的参数设置为默认值，CADNE 参数设置为：$\gamma = 10$，$\beta = 10$，batch-size=128，$\rho = 0.0001$，神经网络各层节点设置如表 2 所示。

Table2Neural Network Structures

表 2 神经网络结构

Dataset	Nodes in each layer
20-NewGroup	1727−1024−128
Cora	2708−1024−128
Citeseer	3312−1024−128
BlogCatalog	5196−1024−128

3.2 聚类实验分析

首先使用CADNE模型得到网络节点的嵌入表示，然后将其运用于节点聚类任务，通过聚类的效果评测网络表示学习的性能。聚类算法采用 K-means，评价标准采用标准互信息（NMI）以及准确率（ACC）。表 3 和表 4 为各算法在数据集的 10 次实验的平均聚类结果。从表中可以观察出，我们提出的 CADNE 模型明显优于其他基线方法，尤其在 20-NEWSGROUP 和 Cora 数据集，准确率提高约 0.2-0.5 不等。M-NMF 也考虑了网络社区特性，但基于矩阵分解的浅层模型，无法捕获网络更高阶复杂结构特性。DEC 通过深度嵌入聚类，在 Blogcatalog 数据表现较好的结果，但缺乏对网络特殊拓扑结构特性的保持，在其他数据集的性能有待提高。SDNE 通过综合考虑网络一阶相似度以及二阶相似度，聚类效果优于采用深度训练模型的 DNGR 算法，但我们提出的 CADNE 模型，在节点低维表示过程中，除了考虑网络的局部以及全局拓扑结构，而且考虑节点聚集的社团结构进行优化，表现出更好的聚类结果，表明了基于网络节点社区结构进行深度嵌入表示的有效性。

Table3NMI of Different Network Embedding on Datasets

表 3 不同网络嵌入方法在数据上的 NMI 值

Algorithm	20NG	BlogCatalog	Citeseer	Cora
DeepWalk	0.001	0.011	0.014	0.004
LINE	**0.608**	0.010	0.003	0.004
SDNE	0.007	0.012	0.007	0.105
DNGR	0.074	0.010	0.014	0.003
M-NMF	0.001	0.013	0.005	0.005
DEC	0.126	**0.218**	0.037	0.073
CADNE	0.587	0.124	**0.079**	**0.198**

Table4ACC of Different Network Embedding on Datasets

表 4 不同网络嵌入方法在数据上的 ACC 值

Algorithm	20NG	BlogCatalog	Citeseer	Cora
DeepWalk	0.348	0.195	0.230	0.205
LINE	0.863	0.191	0.191	0.179
SDNE	0.492	0.200	0.227	0.267
DNGR	0.457	0.192	0.211	0.193
M-NMF	0.347	0.201	0.202	0.189
DEC	0.513	**0.407**	0.266	0.336
CADNE	**0.872**	0.366	**0.281**	**0.393**

3.3 分类实验分析

CADNE 模型得到节点表示之后，将其运用于节点的分类任务，分类结果的好坏可以有效判断网络表示学习模型学习到的嵌入表示是否包含了网络更多的特性。分类算法采用 Liblinear 分类包，采用宏平均（Macro-F1）和微平均（Micro-F1）作为评价指标，随机抽取 10%到 90%的节点嵌入表示作为训练样本，其余作为测试样本。在 20-NewsGroup、Cora、Citeseer、BlogCatalog 数据集的多标签分类结果如表 5-8 所示：

Table 5Multi-label classification results on 20NG dataset

表 5 数据集 20NG 上的分类实验结果

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	DEEPWALK	0.335	0.356	0.341	0.337	0.354	0.353	0.340	0.313	0.329
	LINE	**0.960**	**0.977**	**0.967**	**0.971**	**0.972**	**0.960**	**0.964**	**0.954**	**0.945**
	SDNE	0.717	0.725	0.726	0.721	0.684	0.691	0.687	0.677	0.612
	DNGR	0.873	0.812	0.823	0.815	0.791	0.779	0.827	0.744	0.699
	M-NMF	0.364	0.387	0.328	0.342	0.344	0.336	0.343	0.351	0.340
	DEC	0.515	0.520	0.535	0.520	0.536	0.543	0.528	0.538	0.558
	CADNE	0.867	0.876	0.879	0.868	0.863	0.872	0.871	0.876	0.869
Macro-F1	DEEPWALK	0.333	0.351	0.341	0.335	0.349	0.351	0.326	0.310	0.324
	LINE	**0.960**	**0.977**	**0.968**	**0.971**	**0.972**	**0.959**	**0.964**	**0.954**	**0.945**
	SDNE	0.715	0.724	0.726	0.721	0.684	0.692	0.688	0.678	0.608
	DNGR	0.870	0.813	0.823	0.816	0.791	0.780	0.827	0.744	0.694
	M-NMF	0.347	0.381	0.311	0.317	0.334	0.268	0.340	0.349	0.291
	DEC	0.416	0.501	0.524	0.509	0.493	0.506	0.519	0.528	0.539
	CADNE	0.866	0.877	0.880	0.868	0.864	0.873	0.871	0.877	0.870

Table 6Multi-label classification results onBlogCatalog dataset

表 6 数据集 BlogCatalog 上的分类实验结果

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
Micro-F1	DEEPWALK	0.177	0.223	0.192	0.203	0.192	0.194	0.192	0.195	0.175
	LINE	0.202	0.202	0.215	0.211	0.199	0.189	0.189	0.185	0.186
	SDNE	0.204	0.179	0.189	0.194	0.199	0.194	0.197	0.193	0.186
	DNGR	0.198	0.180	0.205	0.205	0.188	0.175	0.191	0.186	0.184
	M-NMF	0.187	0.211	0.184	0.188	0.188	0.198	0.195	0.187	0.179
	DEC	0.424	0.431	0.446	0.448	0.450	0.444	0.459	0.466	0.481
	CADNE	**0.673**	**0.685**	**0.643**	**0.644**	**0.654**	**0.641**	**0.648**	**0.633**	**0.610**

	DEEPWALK	0.169	0.221	0.186	0.198	0.190	0.190	0.189	0.193	0.172
	LINE	0.199	0.198	0.212	0.205	0.195	0.188	0.184	0.182	0.185
	SDNE	0.196	0.171	0.184	0.189	0.188	0.191	0.192	0.185	0.170
Macro-F1	DNGR	0.194	0.175	0.200	0.203	0.185	0.173	0.188	0.184	0.180
	M-NMF	0.159	0.184	0.161	0.151	0.158	0.178	0.174	0.176	0.175
	DEC	0.404	0.421	0.415	0.443	0.432	0.425	0.450	0.453	0.472
	CADNE	**0.665**	**0.681**	**0.638**	**0.640**	**0.652**	**0.636**	**0.643**	**0.628**	**0.606**

Table 7Multi-label classification results onCiteseer dataset

表 7 数据集 Citeseer 上的分类实验结果

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DEEPWALK	0.255	0.227	0.228	0.224	0.219	0.226	0.209	0.197	0.200
	LINE	0.199	0.186	0.178	0.197	0.193	0.202	0.208	0.205	0.195
	SDNE	0.218	0.247	0.237	0.249	0.228	0.213	0.231	0.221	0.207
Micro-F1	DNGR	0.196	0.252	0.219	0.229	0.219	0.213	0.214	0.210	0.206
	M-NMF	0.175	0.229	0.200	0.212	0.210	0.211	0.213	0.206	0.198
	DEC	0.201	0.248	0.256	0.259	0.257	0.250	0.267	0.268	0.264
	CADNE	**0.405**	**0.376**	**0.352**	**0.363**	**0.360**	**0.345**	**0.360**	**0.348**	**0.323**
	DEEPWALK	0.246	0.192	0.207	0.200	0.194	0.204	0.198	0.179	0.186
	LINE	0.138	0.132	0.133	0.134	0.123	0.143	0.134	0.102	0.148
	SDNE	0.153	0.182	0.185	0.181	0.175	0.169	0.182	0.167	0.168
Macro-F1	DNGR	0.164	0.214	0.182	0.198	0.192	0.186	0.187	0.179	0.180
	M-NMF	0.130	0.176	0.170	0.145	0.175	0.160	0.161	0.163	0.148
	DEC	0.106	0.118	0.124	0.124	0.121	0.131	0.126	0.127	0.154
	CADNE	**0.318**	**0.308**	**0.286**	**0.290**	**0.299**	**0.277**	**0.310**	**0.268**	**0.289**

Table 8Multi-label classification results on Cora dataset

表 8 数据集 Cora 上的分类实验结果

Metric	Algorithm	10%	20%	30%	40%	50%	60%	70%	80%	90%
	DEEPWALK	0.258	0.255	0.242	0.247	0.224	0.214	0.205	0.198	0.209
	LINE	0.295	0.310	0.289	0.300	0.310	0.306	0.296	0.301	0.302
	SDNE	0.347	0.288	0.284	0.283	0.290	0.284	0.291	0.282	0.274
Micro-F1	DNGR	0.269	0.251	0.221	0.250	0.228	0.241	0.221	0.229	0.204
	M-NMF	0.336	0.306	0.303	0.312	0.312	0.296	0.282	0.273	0.240
	DEC	0.359	0.369	0.355	0.362	0.360	0.354	0.352	0.367	**0.410**
	CADNE	**0.406**	**0.408**	**0.402**	**0.420**	**0.414**	**0.420**	**0.403**	**0.396**	0.395
	DEEPWALK	0.107	0.118	0.130	0.140	0.139	0.136	0.135	0.143	0.157
	LINE	0.071	0.079	0.071	0.069	0.077	0.075	0.065	0.070	0.066
	SDNE	0.093	0.070	0.066	0.072	0.069	0.086	0.075	0.075	0.081
Macro-F1	DNGR	0.124	0.125	0.126	0.139	0.126	0.148	0.136	0.140	0.142
	M-NMF	0.098	0.083	0.078	0.091	0.086	0.094	0.085	0.111	0.120
	DEC	0.144	0.148	0.142	0.146	0.147	0.146	0.133	0.153	0.170
	CADNE	**0.353**	**0.341**	**0.331**	**0.349**	**0.341**	**0.332**	**0.298**	**0.333**	**0.249**

由实验结果可见，CADNE 模型分类效果在 BlogCatalog、Citeseer、Cora 数据集上明显优于其他基线方法，其准确率提高 0.1-0.5 不等。结果表明，与基线相比，该方法的学习网络表示能更好地推广到分类任务。其中，CANDE 模型在 BlogCatalog 数据集上优势最为明显，相比基线中传统网络嵌入方法准确率提高约 0.5。这在一定程度上表明我们的 CADNE 模型结构对网络表示学习有积极的影响。在表 6（BlogCatalog）中，当训练百分比从 60%下降到 10%时，我们的方法在基线上的改进幅度更加明显。结果表明，在标记数据有限的情况下，该方法比基线方法有更大的改进。这种优势对于实际应用尤其重要，因为标记的数据通常是稀缺的。在大多数情况下，Deepwalk 性能是网络嵌入方法中最差的，DeepWalk 没有明确的目标函数来捕获网络结构，且所采用的随机游走方式可能引入了噪声，而在大多数情况下，我们的CADNE 模型的性能是网络嵌入方法中最好的，该方法根据节点聚类结构对节点低维表示进行优化，对编码器的编码过程进一步训练，使得节点的低维表示过程保持聚类结构。因此，该方法的学习网络表示能更好地推广到分类任务。

3.4 链接预测实验分析

为了验证CADNE 模型得到的节点低维嵌入在链接预测中的有效性，从低维表示后的样本中随机选取 90%作为训练集，采用逻辑回归分类器进行训练，进行预测，使用 AUC（ROC 曲线下面积）衡量预测的准确性，较高的 AUC 值表示更好的性能。各模型的实验对比结果如表 9 所示：

Table 9 AUC results of different models on networks

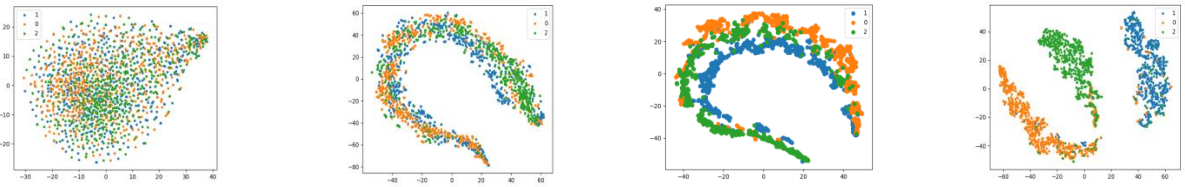
表 9 不同网络嵌入方法在数据集上的 AUC 值

Algorithm\Datasets	20-NewsGroup	Cora	Citeseer	BlogCatalog
DEEPWALK	0.510	0.522	0.556	0.551
LINE	**0.995**	0.537	0.483	0.537
SDNE	0.859	0.472	0.526	0.572
DNGR	0.944	0.506	0.584	0.547
M-NMF	0.517	0.520	0.543	0.543
DEC	0.753	0.582	0.553	0.804
CADNE	0.921	**0.761**	**0.698**	**0.915**

从实验结果可见，相比已有的网络嵌入方法, 我们提出的基于社区优化的深度网络嵌入方法 CADNE 在各数据集上都取得较大提升。具体来说，CADNE 模型在 20-NewsGroup 上 LINE 和 DNGR 取得较好的结果，但 CADNE 与 SDNE 相比提高 0.07，相比其他方法提高约 0.2-0.4。在其他三个数据集 Cora、Citeseer 以及 BlogCatalog 上，我们提出的方法都取得最优结果，比基线方法提高了约 0.1-0.3。以上结果表明本文通过结合节点社团结构的深度网络嵌入方法，能够得到更好的节点低维表示。

3.5 可视化实验分析

为了进一步评测本文提出CADNE 模型节点嵌入表示的有效性，我们在 20-Newsgroup 数据集上与 LINE、 DNGR 以及 SDNE 的可视化结果进行比较。将网络嵌入模型输出得到的节点低维嵌入表示，输入 t-SNE 得到数据样本在 2D 空间的可视化图，其中同颜色的数据点表示同一类别。通过可视化，不同颜色样本点组成的簇间的边界越清晰，说明模型得到的节点表示越好。



LINE　　DNGR　　SDNE　　CADNE

Fig.2Visualization Results of 20-NewsGroup

图 220-NewsGroup 数据集上的可视化结果

从图 2 结果可见，LINE 和 DNGR 的类边界不清晰，而且类内混淆度比较大。尽管 SDNE 能够得到比较好的可视化结果，但不同类的边界也不够清楚。我们提出的 CADNE 则能够得到比较清晰的类边界，三个类间的间距比较大，而且同一个类内相同节点大部分聚在一起。由此可见，通过在节点低维表示过程中，引入节点的聚类结构作为约束，能够得到类边界更加清晰的节点低维表示。

3.6 参数敏感性分析

CADNE 有两个超参数，样本重要度参数 γ 和二阶相似度系数 β。这里选择在四个数据集上进行测试，通过实验分析超参数的选择对 CADNE 模型在链接预测上的性能。除了当前被测试的参数，其他参数均保持默认值。图 3 显示了 γ 取值为 [0,30] 时所有样本数据集 AUC 值的分布情况。从结果可见，当 γ 为 0 时，CADNE 取得的效果最差。此时相当于 CADNE 模型仅利用了网络拓扑结构中的一阶近邻信息，无法完全保留网络中高阶的相似度；随着 γ 增大，CADNE 模型的效果先迅速提升，在 γ =10 时达到最好之后缓慢下降，在 Cora 和 Citeseer 数据集上结果比较稳定。因此，本实验设置中该参数设置为 10。在图 4 中，我们可以得到类似的结果，当 β 从 1 增至 30 过程中，开始 CADNE 性能迅速提升，到 10 取得最优结果，之后缓慢下降。具体地，在 β =1 时效果最差，此时将邻接矩阵中零元素与非零元素同等对待进行模型训练，因此会重建更多的零元素，引入的噪声信息影响了最终节点嵌入表示的性能。随着 β 增加，模型会倾向于重建更多的非零元素，因此效果有显著提升。但过大的 β 使得忽略零元素的重建过程，性能会降低，因此在实验过程中 β 设置为 10。

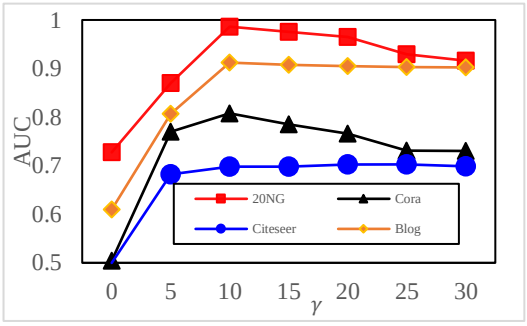


Fig.3 AUC values of different γ

图 3 不同参数 γ 的 AUC 值

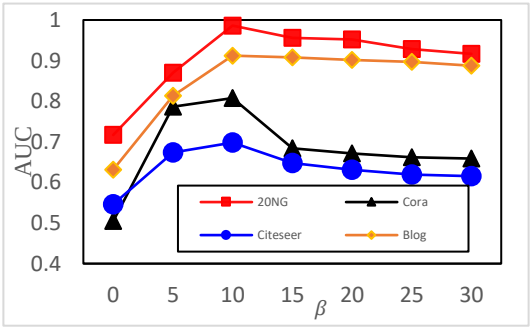


Fig.4AUC values of differentβ

图 4 不同参数β的 AUC 值

4.总结

本文提出了一种基于社区结构优化的网络表示方法，在节点低维表示过程中，不仅保持网络的局部和全局拓扑结构关系，而且融合节点潜在的社区特性。打破传统的网络表示学习方法局限，得到低维稠密更具有表示能力的特征表示。本文提出的基于社区结构优化进行节点低维特征表示的深度自编码聚类模型 CADNE 能够同时学习节点的低维表示和节点所属社区的指示向量。在多个数据集上与已有经典网络表示学习方法对比实验表明：CADNE 方法具有较好的节点低维表示能力。

参考文献

[1] Bothorel C, Cruz JD, Magnani M. Clustering attributed graphs: models, measures and methods[J]. Network Science, 2015, 3(3): 408-444.

[2] Qi Jinshan, Liang Xun, Li Zhiyu,ChenYanfang, Xu Yuan. Representation learning of large-scale complex information network: concepts, methods and challenges[J]. Chinese Journal of Computers, 2018,41(10):2394-2420(in Chinese)
(齐金山, 梁循, 李志宇, 陈燕方, 许媛. 大规模复杂信息网络表示学习:概念、方法与挑战[J]. 计算机学报, 2018, 41(10): 2394-2420)

[3] Yin Ying, Ji Lixin, Huang Ruiyang, Du Lixin. Research and development of network representation learning[J]. Chinese Journal of Network and Information Security,2019,5(2):77-87(in Chinese)
(尹赢, 吉立新, 黄瑞阳, 杜立新. 网络表示学习的研究与发展[J].网络与信息安全学报,2019,5(2):77-87)

[4] Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding[J]. Science, 2000, 290: 2323-2326.

[5] Belkin M, Niyogi P. Laplacian eigenmaps and spectral techniques for embedding and clustering[C]// Proc of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic. Cambridge: MIT Press, 2001:585-591

[6] Shaw B, Jebara T, Structure preserving embedding[C]//Proc of the 26th Annual International Conference on Machine Learning. New York: ACM, 2009:937–944.

[7] Ahmed A, Shervashidze N, Narayanamurthy S, *et al.*, Distributed large-scale natural graph factorization[C]//Proc of the 22nd international conference on World Wide Web. New York: ACM, 2013:37-48.

[8] Cao S, Lu W, Xu Q, Grarep: Learning graph representations with global structural information[C]//Proc of the 24th ACM International on Conference on Information and Knowledge Management. New York: ACM, 2015: 891-900.

[9] Ou M, Cui P, Pei J, *et al.*, W. Zhu, Asymmetric transitivity preserving graph embedding[C]// Proc of ACM SIGKDD. New York:ACM,2016: 1105-1114.

[10] Wang X, Cui P, Wang J, et al., Community Preserving Network Embedding[C]// Proc of the Thirty-Fist AAAI Conference on Artificial Intelligence, San Francisco: AAAI Press, 2017: 203-209.

[11] Xu Q, et al. Deep neural networks for learning graph representations[C]// Proc of the Thirtieth AAAI Conference on Artificial Intelligence. Palo Alto:AAAI Press, 2016:1145-1152.

[12] Perozzi B, Al-Rfou R, Skiena S, Deepwalk: Online learning of social representations[C]// Proc of 20th international conference on Knowledge discovery and data mining. New York:ACM,2014:701-710.

[13] Grover A, Leskovec J, node2vec: Scalable feature learning for networks[C]// Proc of the 22nd International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2016: 855-864.

[14] Chen H, Perozzi B, Hu Y, et al. HARP: Hierarchical Representation Learning for Networks[C]// Proc ofNationalConference on Artificial Intelligence. Palo Alto: AAAI Press, 2018: 2127-2134.

[15] Li J, Zhu J, Zhang B, Discriminative deep random walk for network classification[C]//Proc of the 54th Annual Meeting of the Association for Computational Linguist. Berlin: ACL,2016: 1004-1013.

[16] Chen Li, Zhu Peisong, Qian Tieyun, Zhu Hui, Zhou Jing. Edge sampling based network embedding model[J]. Journal of Software, 2018, 29(3): 756-771(in Chinese)
(陈丽，朱裴松，钱铁云，朱辉，周静．基于边采样的网络表示学习模型[J]．软件学报，2018，29(3)：756-771)

[17] Wang D, Cui P, Zhu W, Structural deep network embedding[C]//Proc of the 22nd International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2016: 1225-1234.

[18] Cao S, Lu W, Xu Q, Deep neural networks for learning graph representations[C]//Proc of the Thirtieth AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2016: 1145-1152.

[19] Xie J, Girshick R, Farhadi A, et al, Unsupervised deep embedding for clustering analysis[C]// Proc of the 33rd International Conference on Machine Learning. New York: JMLR.org，2016: 478-487.

[20] http://qwone.com/ jason/20Newsgroups/

[21] http://vmwxs.umd.edu/projects/linqs/projects/lbc/index.html

[22] http://citeseer.ist.psu.edu/

[23] Tang L, Liu H, Relational learning via latent social dimensions[C]//Proc of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. New York: ACM, 2009: 817-816.