文章编号:1672-3961(2025)01-0041-10 DOI:10.6040/j.issn.1672-3961.0.2024.191

基于双视角网络嵌入聚类集成社区发现算法

王英楠[1],郑文萍[2,3*],杨贵[2]
(1.山西医科大学汾阳学院卫生信息管理系, 山西 汾阳 032200; 2.山西大学计算机与信息技术学院, 山西 太原 030006;
3.计算智能与中文信息处理教育部重点实验室(山西大学), 山西 太原 030006)

摘要:针对现有网络嵌入方法忽略高阶结构,嵌入过程与社区发现任务独立进行,影响社区发现质量的问题,提出基于双视角网络嵌入聚类集成社区发现算法(community detection algorithm based on dual-view network embedded clustering integration, DNECI),算法包括双视角网络嵌入和聚类集成两部分。双视角网络嵌入模块对网络属性信息与拓扑信息实现自适应融合,保留网络属性信息与拓扑的高阶结构。聚类集成模块包括模块度优化和聚类优化两个组件,模块度优化组件利用高阶拓扑结构得到具有最优模块度的社区结果;聚类优化组件通过自监督聚类方法在嵌入空间得到聚类结果;引入互监督机制使两种视角的社区发现结果具有一致性。在 4 个真实数据集与 15 个算法进行对比试验,结果表明,DNECI 在准确率和标准互信息至少比最先进的基准算法平均提高 2.5%和 1.4%,在调整兰德系数和 $F1$ 分数至少平均提高 3.7%和 1.7%,具有较好的社区发现效果。

关键词:社区发现;网络嵌入;模块度;自监督;高阶结构
中图分类号:TP391 **文献标志码**:A
引用格式:王英楠,郑文萍,杨贵. 基于双视角网络嵌入聚类集成社区发现算法[J]. 山东大学学报(工学版),2025,55(1):41-50.
WANG Yingnan, ZHENG Wenping, YANG Gui. Community detection algorithm based on dual-view network embedded clustering integration[J]. Journal of Shandong University (Engineering Science), 2025, 55(1):41-50.

Community detection algorithm based on dual-view network embedded clustering integration

WANG Yingnan[1], ZHENG Wenping[2,3*], YANG Gui[2]
(1. Fenyang College of Shanxi Medical University, Fenyang 032200, Shanxi, China; 2. School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China; 3. Key Laboratory of Computation Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan 030006, Shanxi, China)

Abstract: In response to the issues where existing network embedding methods neglected higher-order structures, and the embedding process was conducted independently of the community detection task, which affected the quality of community detection, a community detection algorithm based on dual-view network embedded clustering integration (DNECI) was proposed. The algorithm consisted of two parts: dual-view network embedding and clustering integration. The dual-view network embedding module adaptively fused network attribute information with topological information, preserving the higher-order structures of both. The clustering integration module included two components: modularity optimization and clustering optimization. The modularity optimization component used higher-order topological structures to achieve community results with optimal modularity, while the clustering optimization component obtained clustering results in the embedding space through a self-supervised clustering method. A mutual supervision mechanism was introduced to ensure consistency between the community detection results from both perspectives. Comparative experiments on 4 real datasets and 15 algorithms showed that DNECI improved accuracy and normalized mutual information by at least 2.5% and 1.4% on average compared to state-of-the-art benchmark algorithms, and improved the adjusted Rand index and $F1$ score by at least 3.7% and 1.7% on average, demonstrating better community detection performance.

Keywords: community detection; network embedding; modularity; self supervise; higher-order structure

收稿日期:2024-07-30
基金项目:国家自然科学基金资助项目(62072292);山西省 1331 工程资助项目
第一作者简介:王英楠(1995—),男,山西大同人,助教,硕士,主要研究方向为图神经网络、聚类分析。E-mai:1291533544@qq.com
*通讯作者简介:郑文萍(1979—),女,山西榆次人,教授,博士生导师,博士,主要研究方向为复杂网络分析、生物信息学、聚类分析。
E-mail: wpzheng@sxu.edu.cn

0 引言

现实世界中的许多复杂系统可以表示为复杂网络,一组节点通过相对紧密连接形成网络模块(也称为社区结构)行使系统功能,如在基因疾病网络中,同一社区的基因会引起同种疾病;在线购物网络中,同一社区的商品具有类似用途。研究复杂网络社区结构对于理解复杂系统结构和功能至关重要。

复杂网络通常伴随着节点属性数据维度高且节点之间拓扑连接稀疏问题,对复杂网络进行低维嵌入后进行分析会缓解稀疏问题。早期对网络进行嵌入算法如基于随机游走的网络嵌入、矩阵分解网络嵌入和基于深度学习网络嵌入方法大都利用网络拓扑结构获得网络的表示,忽略了网络属性信息;现有基于网络嵌入进行社区发现的方法通常只保留网络低阶结构,忽略了网络结构蕴含的高阶拓扑关系;网络嵌入过程与社区发现任务独立进行,影响了社区发现质量;基于图神经网络嵌入方法核心思想是邻域聚合,利用网络邻接矩阵实现属性信息聚合,本质是对网络属性特征聚合,网络拓扑信息作为辅助信息指导哪些属性特征进行聚合,网络拓扑本身蕴含的高阶信息未有效挖掘。

本研究提出基于双视角网络嵌入聚类集成社区发现算法(community detection algorithm based on dual-view network embedded clustering integration, DNECI),将网络领域属性视角和网络拓扑视角信息有效融合,网络嵌入与社区发现耦合进行,从嵌入空间分布角度发现社区结果和从拓扑连接角度发现社区结果集成得到最终社区。算法 DNECI 包括双视角网络嵌入和聚类集成两部分。双视角网络嵌入模块对网络属性信息与拓扑信息自适应融合,同时保留了网络属性信息与拓扑高阶结构。聚类集成模块由模块度优化和聚类优化两个组件构成,模块度优化组件从高阶拓扑结构出发上得到具有最优模块度的社区结果;聚类优化组件在嵌入空间上通过自监督聚类得到聚类结果;引入互监督机制,保证了两种社区结果协同,提高了社区发现结果质量。

1 相关工作

1.1 传统社区发现

传统社区发现算法大多利用网络拓扑信息来检测社区结构,有代表性的是基于模块度优化方法、基于标签传播方法。

模块度 Q 能够度量一个网络拓扑结构是否具有明显社区结构,Q 越高,社区结构越明显。很多经典的社区发现算法是基于模块度优化的。BGLL(blondel-guillaume-lambiotte-lefebvre)[1]算法、FMM(fast modularity maximization)[2]算法、CNM(clause-newman-moore)[3]算法都是基于模块度优化社区发现算法。将社区结构模块度作为优化目标,得到社区结构显著结果,社区内部节点连边稠密,社区之间连边稀疏。此类算法在每次迭代中都需要知道网络全局信息,对于大规模网络计算代价较大。

基于标签传播的算法将领域节点中出现次数最多的标签作为当前节点标签,标签更新过程迭代进行,直至节点标签不再变化。如 LPA(label propagation algorithm)[4], LPAm(modularity-specialized label algorithm)[5]。基于标签传播算法仅根据节点直接邻域信息来更新标签,收敛速度快,有接近线性时间的复杂度,在标签更新过程中有较大的随机性,导致社区发现结果不具有稳定性。

1.2 网络嵌入

网络嵌入旨在从网络拓扑结构及属性信息中学习低维向量表示,可有效应对大规模网络稀疏性问题,有助于理解节点间语义关联。早期嵌入方法通常利用网络拓扑结构学习节点嵌入,LE(laplacian eigenmaps)[6]、Isomap(isometric mapping)[7]、LLE(locally linear embedding)[8]等在保留局部流形结构的同时得到数据嵌入,涉及到矩阵分解操作,计算代价高,不适用于大规模网络。文献[9]提出 Deep Walk 算法,以当前节点为起点进行随机游走,根据游走路径上节点属性信息学习得到节点嵌入。文献[10]提出了 Node2Vec 算法,在随机游走过程中增加了搜索偏置调节游走深度与宽度,获取节点局部结构信息。文献[11]提出了 SDNE(structural deep work embedding)通过使用深度自编码模型学习保持一阶相似性和二阶相似性得到网络嵌入。文献[12]提出 DNGR(deep neural networks for learning graph representation)将随机游走与深度自编码器相结合,利用随机游走得到节点相似性矩阵,用堆叠自编码器学习节点嵌入。文献[13]提出 M-NMF(modularized nonnegative matrix factorization)对模块化矩阵进行非负矩阵分解,将社区的结构信息融入到网络嵌入中。

以上网络嵌入算法都是基于网络拓扑结构进行节点嵌入,将节点属性与网络拓扑两个视角信息结合起来有利于学习到更好的向量表示。文献[14]提出 TADW(text-associated deep walk)算法,对概率转移矩阵进行分解,添加节点文本向量矩阵,将拓扑特征与属性特征联合考虑。文献[15]提

出 AANE(accelerated attributed network embedding)算法,通过属性信息生成余弦相似度,利用节点属性关联矩阵得到网络嵌入。文献[16]提出图卷积神经网络(graph convolutional networks, GCN)迭代聚合邻居节点特征以更新节点特征来实现半监督节点分类。图注意力网络(graph attention networks, GAT)[17]考虑节点邻居的不同影响力获得更具有表达能力的嵌入表示。文献[18]提出图自动编码器模型(graph auto-encoders, GAE),利用 GCN 作为编码器得到节点嵌入,重构邻接矩阵尽可能保持网络拓扑结构。文献[19]提出图注意力自动编码器模型(graph attention networks auto-encoders, GATE),根据邻居节点影响力对邻居节点的特征加权以更新中心节点特征得到节点嵌入。

1.3　基于网络嵌入的社区发现

得到节点嵌入后,运用聚类算法进行聚类发现社区。文献[12]提出 DNGR 算法得到节点嵌入,利用 k-means 算法进行社区发现。网络嵌入与节点聚类分开进行,得到的网络嵌入强调保持原始网络低阶相似性,忽略了子图、社区结构等更高阶关系,不容易发现内部连接比较稀疏社区。

深度聚类近些年受到很多研究者关注,文献[20]提出深度聚类算法(deep embedded clustering, DEC);文献[21]提出深度注意力图聚类算法(deep attentional embedded graph clustering, DAEGC);文献[22]提出结构深度聚类算法(structural deep clustering network, SDCN);文献[23]提出 DSNE(dual supervised network embedding based community detection algorithm)。这些深度聚类算法通过最小化重构损失与聚类损失使得嵌入学习过程与聚类耦合进行,得到内聚性比较高的嵌入表示,提高了聚类表现。

现有图深度聚类得到节点嵌入后,利用节点相似距离进行聚类,属性相似节点在嵌入空间内距离较近,聚类过程中对拓扑结构紧密性考虑较少。利用图神经网络进行网络嵌入,邻接矩阵作为网络拓扑结构信息来源,指导属性特征聚合,本质仍是对属性信息处理,拓扑结构本身蕴含的丰富信息未被有效挖掘。本研究提出 DNECI 算法,从网络拓扑与属性两个不同视角信息进行有效融合,对网络高阶信息进行了挖掘与保留。

2　DNECI 算法

给定一个图 $G=(V,E,\boldsymbol{X})$,V 是节点的集合,E 是节点之间边的集合,$\boldsymbol{A}\in\mathbf{R}^{n\times n}$ 是图 G 的邻接矩阵,n 表示节点数量,$\boldsymbol{X}\in\mathbf{R}^{n\times m}$ 是节点属性矩阵,其中 m 是属性数量。

DNECI 算法包括双视角嵌入模块、聚类集成模块 2 部分,算法框架如图 1 所示。$\hat{\boldsymbol{A}}$ 是重构邻接矩阵,$\boldsymbol{Z}_a$ 是属性嵌入,$\boldsymbol{Z}_s$ 是结构嵌入,$\boldsymbol{Z}_f$ 是融合嵌入,$\hat{\boldsymbol{X}}$ 为重构属性矩阵。

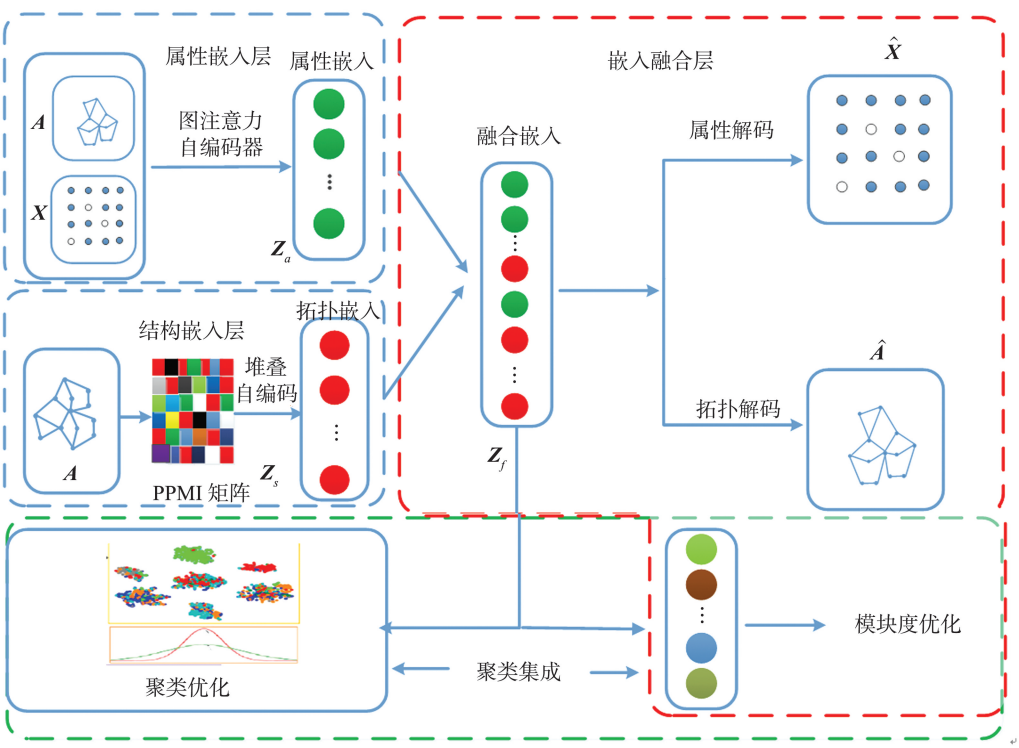


图 1　基于双视角网络嵌入的聚类集成社区发现算法框架

Fig.1　Community detection algorithm based on dual-view network embedded clustering integration

聚类集成模块由模块度优化和聚类优化两个组件构成。模块度优化组件通过最大化模块度保留网络高阶结构信息,连接紧密的节点构成社区,得到社区结构显著的社区结果;聚类优化组件从嵌入空间出发,采用自监督聚类方法优化聚类结果,使得同一社区类节点在嵌入空间更相似。引入互监督机制对两种组件集成,使两种聚类结果一致,保证同一社区的节点在拓扑结构与嵌入空间上都有较高内聚性。

2.1 属性嵌入层

属性嵌入层采用图注意力自编码器学习节点低维表示,将不同邻居节点对中心节点不同影响力考虑在内,突出了与中心节点相关信息,设在第 l 层节点 i 对应的特征向量为 $\boldsymbol{z}_i^{(l)}$,衡量不同邻居的影响力差异对邻居的嵌入进行聚合运算更新节点嵌入 $\boldsymbol{z}_i^{(l+1)}$:

$$\boldsymbol{z}_i^{(l+1)} = \sigma\Big(\sum_{j\in N_i}\alpha_{ij}^{(l)}\boldsymbol{z}_j^{(l)}\boldsymbol{W}^{(l)}\Big), \tag{1}$$

式中,N_i 是节点 i 的邻居节点集,σ 是激活函数,$\boldsymbol{W}^{(l)}$ 为第 l 层注意力网络中的权重矩阵。$\alpha_{ij}^{(l)}$ 是注意力系数,表明第 l 层节点 j 对节点 i 的重要性,定义如下:

$$\alpha_{ij}^{(l)} = \frac{\exp\left(\text{LeakyReLU}\left(\boldsymbol{a}^{\mathrm{T}}\left[\boldsymbol{z}_i^{(l)}\boldsymbol{W}^{(l)} \,\|\, \boldsymbol{z}_j^{(l)}\boldsymbol{W}^{(l)}\right]\right)\right)}{\sum_{k\in N_i}\exp\left(\text{LeakyReLU}\left(\boldsymbol{a}^{\mathrm{T}}\left[\boldsymbol{z}_i^{(l)}\boldsymbol{W}^{(l)} \,\|\, \boldsymbol{z}_k^{(l)}\boldsymbol{W}^{(l)}\right]\right)\right)}, \tag{2}$$

式中,$\|$ 表示向量拼接,$\boldsymbol{a}$ 为权重向量,LeakyReLU 为非线性激活函数,通过 softmax 得到归一化注意力系数 $\alpha_{ij}^{(l)}$。

采用 2 层图注意力网络提取节点属性信息,令节点属性特征 $\boldsymbol{z}_i^{(0)} = \boldsymbol{x}_i$,代入式(1),可以得到:

$$\boldsymbol{z}_i^{(1)} = \sigma\Big(\sum_{j\in N_i}\alpha_{ij}\boldsymbol{x}_j\boldsymbol{W}^{(0)}\Big), \tag{3}$$

$$\boldsymbol{z}_i^{(2)} = \sigma\Big(\sum_{j\in N_i}\alpha_{ij}\boldsymbol{z}_j^{(1)}\boldsymbol{W}^{(1)}\Big), \tag{4}$$

将第二层注意力网络输出的节点表示作为经过属性嵌入层提取得到的属性嵌入向量 $\boldsymbol{Z}_a$:

$$\boldsymbol{Z}_a = \boldsymbol{Z}^{(2)} = (\boldsymbol{z}_1^{(2)}\ \boldsymbol{z}_2^{(2)}\ \cdots\ \boldsymbol{z}_n^{(2)}), \tag{5}$$

2.2 结构嵌入层

邻接矩阵表示网络节点间直接相邻关系,随着网络规模增大,邻接矩阵表现出明显稀疏性,即节点直接邻居数远小于邻接矩阵维度,网络低阶结构不足以表示节点间相关性,网络节点间拓扑相关性不仅包含节点间低阶相关性信息,还应包含多步邻域内关联的高阶相关性。

结构嵌入层对网络低阶与高阶拓扑相关性进行提取,旨在得到网络中节点包含高阶拓扑相关性的结构嵌入,将节点间拓扑相关性表示到低维空间。此处采用重启的随机游走得到图 G 的 u 步概率转移矩阵 $\boldsymbol{M}^{(u)}$ 表示图 G 中节点间的 u 步到达的概率,公式如下:

$$\boldsymbol{M}^{(u)} = (1-\mu)\boldsymbol{M}^{(u-1)}\boldsymbol{T} + \mu\boldsymbol{I}, \tag{6}$$

式中,$\boldsymbol{I}$ 是单位矩阵,$\boldsymbol{T} = \boldsymbol{D}^{-1}\boldsymbol{A}$,$\boldsymbol{D}$ 是度矩阵,μ 为随机游走的重启概率。令 $\widetilde{\boldsymbol{M}}^{(U)} = \sum_{r=1}^{U}\boldsymbol{M}^{(u)}$,表示网络中节点间的 U 步内的游走概率。

游走概率矩阵 $\widetilde{\boldsymbol{M}}^{(U)}$ 表示节点间连接紧密程度,使网络中度大的节点间连接更紧密,为了消除大度节点对节点间相关性影响,将游走概率矩阵 $\widetilde{\boldsymbol{M}}^{(U)}$ 转换为 PPMI 矩阵表示网络中节点间的 U 步关联关系,公式如下:

$$\boldsymbol{C}_{\text{PPMI}} = \max\left(\text{lb}\left(\frac{\widetilde{\boldsymbol{M}}^{(U)}\boldsymbol{\Theta}}{\text{col}(\widetilde{\boldsymbol{M}}^{(U)})\,\text{row}(\widetilde{\boldsymbol{M}}^{(U)})}\right), 0\right), \tag{7}$$

式中,$\boldsymbol{C}_{\text{PPMI}}$ 为 PPMI 矩阵,$\boldsymbol{\Theta}$ 是 $\widetilde{\boldsymbol{M}}^{(U)}$ 中所有元素之和,col($\widetilde{\boldsymbol{M}}^{(U)}$)对 $\widetilde{\boldsymbol{M}}^{(U)}$ 行求和所得的列向量,row($\widetilde{\boldsymbol{M}}^{(U)}$)是对 $\widetilde{\boldsymbol{M}}^{(U)}$ 中列元素求和所得的行向量。

PPMI 矩阵体现网络中任意两个节点间 U 步邻域内拓扑相关性,大规模网络中,PPMI 矩阵有较高稀疏性,利用堆叠自编码良好的数据压缩表达能力,将 $\boldsymbol{C}_{\text{PPMI}}$ 包含的拓扑结构信息压缩至低维稠密向量空间,得到网络拓扑结构嵌入表示 $\boldsymbol{Z}_s$。

2.3 嵌入融合层

拓扑连接紧密且属性相似性高的节点在嵌入空间上应更接近。构造网络拓扑连接时,节点间存在假阳性或假阴性连接,导致拓扑连接角度紧密程度与属性信息相似性匹配度低;属性信息高维稀疏性或部分属性缺失现象,仅利用属性信息不能充分挖掘网络节点间相似性。

对两种不同视角信息有效融合进行社区发现,使网络拓扑结构与节点属性信息相互补充,更清晰识别社区边界,提高社区发现质量。本节构建基于自编码的嵌入融合层,引入模块度最优化损失,自适应有效融合属性信息与拓扑信息。

将属性嵌入层得到的属性嵌入 $\boldsymbol{Z}_a$ 和结构嵌入层得到的结构嵌入 $\boldsymbol{Z}_s$ 拼接得到融合嵌入 $\boldsymbol{Z}_f$:

$$\boldsymbol{Z}_f = [\boldsymbol{Z}_a \,\|\, \boldsymbol{Z}_s], \tag{8}$$

式中,$\|$ 表示对向量的拼接,为实现属性嵌入 $\boldsymbol{Z}_a$ 和拓扑嵌入 $\boldsymbol{Z}_s$ 的有效互补,分别构造属性解码器和拓扑解码器对融合嵌入 $\boldsymbol{Z}_f$ 进行解码。

属性解码器采用 2 层图注意力网络重构节点属性信息,解码过程如下:

$$\hat{\boldsymbol{z}}_i^{(1)} = \sigma\Big(\sum_{j\in N_i}\hat{\alpha}_{ij}^{(1)}\hat{\boldsymbol{z}}_j^{(0)}\hat{\boldsymbol{W}}^{(1)}\Big), \tag{9}$$

$$\hat{\boldsymbol{z}}_i^{(2)} = \sigma\Big(\sum_{j\in N_i}\hat{\alpha}_{ij}^{(2)}\hat{\boldsymbol{z}}_j^{(1)}\hat{\boldsymbol{W}}^{(2)}\Big), \tag{10}$$

式中,$\hat{\boldsymbol{z}}_i^{(0)}=\boldsymbol{z}_f$, $\hat{\boldsymbol{W}}^{(1)}$、$\hat{\boldsymbol{W}}^{(2)}$ 为权重参数,$\hat{\alpha}_{ij}^{(1)}$、$\hat{\alpha}_{ij}^{(2)}$ 是节点 i 和 j 的注意力系数。解码器输出 $\hat{\boldsymbol{z}}_i^{(2)}$ 是节点 i 的重构属性 $\hat{\boldsymbol{x}}_i$,公式如下:

$$\hat{\boldsymbol{x}}_i=\hat{\boldsymbol{z}}_i^{(2)}, \tag{11}$$

最小化 $\hat{\boldsymbol{x}}_i$ 与 $\boldsymbol{x}_i$ 之间的差距,属性重构损失 L_a 函数如下:

$$L_a=\sum_{i=1}^{N}\|\boldsymbol{x}_i-\hat{\boldsymbol{x}}_i\|_2。 \tag{12}$$

拓扑解码器采用内积解码重构网络邻接矩阵,内积解码构造如下:

$$\hat{\boldsymbol{A}}=\mathrm{sigmoid}(\boldsymbol{Z}_f^{\mathrm{T}}\boldsymbol{Z}_f), \tag{13}$$

式中,sigmoid(·) 是激活函数。采用交叉熵损失作为邻接矩阵重构损失 L_a,公式如下:

$$L_a=-\frac{1}{N}\sum \boldsymbol{A}_{ij}\mathrm{lb}\hat{\boldsymbol{A}}_{ij}+(1-\boldsymbol{A}_{ij})\mathrm{lb}(1-\hat{\boldsymbol{A}}_{ij})。 \tag{14}$$

社区结构是一种网络高阶拓扑信息,其内部节点间有更高相似度。在节点嵌入中引入社区结构,使网络嵌入保留节点高阶拓扑信息。模块度是度量社区发现质量的重要指标,模块度越高,越能呈现出“低耦合、高内聚”的特点。优化模块度指标得到保留了原始网络高阶社区结构的节点嵌入。

在融合嵌入 $\boldsymbol{Z}_f$ 上利用全连接层构成评估分类网络,通过 softmax 得到节点对社区的归属度矩阵 $\boldsymbol{H}$,公式如下:

$$\boldsymbol{H}=\mathrm{softmax}(\boldsymbol{Z}_f\boldsymbol{W}_f), \tag{15}$$

式中 $\boldsymbol{W}_f$ 为评估分类网络的权重参数。

计算 $\boldsymbol{H}$ 对应的社区发现结果的模块度,方法如下:

$$M_{\mathrm{od}}=\frac{1}{4m}\mathrm{Tr}(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{H}), \tag{16}$$

式中,$\boldsymbol{B}$ 是模块度矩阵,其元素 $B_{ij}=A_{ij}-d_id_j/2m$,Tr(·) 是矩阵的迹,d_i 是节点 i 的度。

为得到具有最优模块度的社区发现结果,模块度损失如下:

$$L_{\mathrm{mod}}=-\frac{1}{4m}\mathrm{Tr}(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{B}\boldsymbol{H})。 \tag{17}$$

嵌入融合层通过联合优化 L_a、L_s 和 L_{mod},拓扑视角和属性视角信息自适应融合,最大程度保留了网络的节点属性、低阶拓扑连接和高阶社区结构信息。嵌入融合层损失函数 L_f 定义如下:

$$L_f=L_a+L_s+\alpha L_{\mathrm{mod}}, \tag{18}$$

式中,α 为 L_{mod} 的权重超参数,联合优化 L_f 得到属性与拓扑视角自适应融合的网络嵌入 $\boldsymbol{Z}_f$。

2.4　聚类集成

从拓扑角度,社区原始定义便是基于网络拓扑链接紧密程度。在 2.3 节中,优化模块度,使嵌入保持了社区显著性,蕴含了高阶结构信息,得到社区分配结果 $\boldsymbol{H}$。

聚类优化模块从嵌入分布角度出发,采用文献[21]提出的方法,联合优化聚类结果及节点表示。根据节点与聚类中心距离得到节点和社区中心聚类软分配,从聚类软分配构造高置信度辅助目标分配来细化聚类中心,通过优化聚类软分配与辅助目标分配两个分布之间差异,提高聚类内聚性。

在网络融入嵌入矩阵上执行 k-means 聚类获得初始社区中心 $\boldsymbol{\mu}=(\boldsymbol{\mu}_1\ \cdots\ \boldsymbol{\mu}_k)$,得到聚类软分配,公式如下:

$$q_{ik}=\frac{(1+\|\boldsymbol{z}_f^i-\boldsymbol{\mu}_k\|^2/\varphi)^{-\frac{\varphi+1}{2}}}{\sum_{k'}(1+\|\boldsymbol{z}_f^i-\boldsymbol{\mu}_{k'}\|^2/\varphi)^{-\frac{\varphi+1}{2}}}, \tag{19}$$

式中,q_{ik} 表示节点 i 分配到社区 k 的概率,α 是 t 分布的自由度,设 $\varphi=1$,引入辅助目标分布 $\boldsymbol{P}$:

$$p_{ik}=\frac{q_{ik}^2/\sum_i q_{ik}}{\sum_{k'}(q_{ik'}^2/\sum_i q_{ik})}, \tag{20}$$

聚类优化损失为类分布 $\boldsymbol{Q}$ 和辅助分布 $\boldsymbol{P}$ 之间的相对熵(kullback-leibler divergence, KL)公式如下,

$$L_c=\mathrm{KL}(\boldsymbol{P}\|\boldsymbol{Q})=\sum_i\sum_k p_{ik}\mathrm{lb}\frac{p_{ik}}{q_{ik}}。 \tag{21}$$

模块度优化所得聚类结果 $\boldsymbol{H}$ 和自监督聚类结果 $\boldsymbol{P}$ 分别从拓扑连接视角和嵌入空间视角进行社区发现,导致两个视角的社区结果有差异,优化过程中导致过早收敛使得结果次优。引入互监督机制使两种聚类结果协同一致,根据两个聚类分布的相对熵定义聚类一致性损失,公式如下。

$$L_{\mathrm{mc}}=\frac{1}{2}(\mathrm{KL}(\boldsymbol{H}\|\boldsymbol{P})+\mathrm{KL}(\boldsymbol{P}\|\boldsymbol{H})=$$

$$\frac{1}{2}(\sum_i\sum_k h_{ik}\mathrm{lb}\frac{h_{ik}}{p_{ik}}+\sum_i\sum_k p_{ik}\mathrm{lb}\frac{p_{ik}}{h_{ik}})。 \tag{22}$$

通过聚类集成模块,对两种社区发现方法进行联合优化,得到质量更好的社区划分结果。最后以辅助目标分配的社区 $\boldsymbol{P}$ 作为社区划分结果。

2.5　模型训练

DNECI 算法通过堆叠自编码对网络 PPMI 矩阵进行压缩,得到初始结构嵌入,将初始结构嵌入和属性嵌入融合,通过优化重构损失和最大化模块度完成预训练,使嵌入不仅保留网络高阶拓扑结构,还保留原始网络属性信息。公式如下:

$$L_{\mathrm{pre}}=L_a+L_s+\alpha L_{\mathrm{mod}}。 \tag{23}$$

预训练后,采用 k-means 得到聚类优化模块需要的初始社区中心。采用模块度优化和聚类优化构成的聚类集成进行社区发现。公式如下:

$$L=L_f+\beta L_c+\gamma L_{\mathrm{mc}}=L_a+L_s+\alpha L_{\mathrm{mod}}+\beta L_c+\gamma L_{\mathrm{mc}},\quad(24)$$

式中 α、β、γ 是权衡不同损失权重的超参数。将聚类优化的节点的辅助目标分配作为节点 i 的社区标签,公式如下:

$$s_i=\arg\max_k p_{ik}。\quad(25)$$

2.6 算法描述

算法 1 DNECI 算法。

输入 图 $G=(V, E, X)$,社区数量 K,参数 α、β、γ, PPMI 最大预训练迭代次数 T_1,模型最大预训练迭代次数 T_2,模型最大联合训练迭代次数 T_3。

输出 节点的社区标签分配 $\boldsymbol{S}$。

① // PPMI 预训练

② 根据式(6)利用重启的随机游走对网络进行遍历,根据式(7)得到包含网络中节点拓扑相关性的 PPMI 矩阵;

③ for epoch in 0, 1, …, T_1 do

④ 利用堆叠自编码器对 PPMI 矩阵进行预训练,得到结构嵌入 $\boldsymbol{Z}_s$;

⑤ end

⑥ // 模型预训练

⑦ 利用属性嵌入层对网络属性进行提取,得到网络属性视角的嵌入 $\boldsymbol{Z}_a$;

⑧ 将 PPMI 预训练好的结构嵌入 $\boldsymbol{Z}_s$ 与属性嵌入 Z 融合得到初始的融合嵌入 $\boldsymbol{Z}_f$;

⑨ for epoch in 0, 1, …, T_2 do

⑩ 利用式(23)对模型及进行预训练,得到网络属性信息和拓扑高阶信息的融合嵌入 $\boldsymbol{Z}_f$;

⑪ end

⑫ 在融合嵌入 $\boldsymbol{Z}_f$ 上利用 k-means 得到 K 个初始社区中心;

⑬ // 模型联合训练

⑭ for epoch in 0, 1, …, T_3 do

⑮ 计算模块度损失;

⑯ 计算自监督聚类损失;

⑰ 计算聚类集成损失;

⑱ 据式(24)更新模型中涉及到的参数;

⑲ end

⑳ 计算节点 i 的社区标签 $s_i=\arg\max_k p_{ik}$,返回 $\boldsymbol{S}=\{s_i\mid 1\leqslant i\leqslant n\}$,算法结束。

3 试验设计与结果分析

3.1 数据集及对比算法

本研究在 Cora、ACM、Citeseer 和 DBLP 4 个带标签真实网络数据集上进行对比试验,基准算法中,LPA、Infomap、BGLL 等传统算法利用拓扑结构直接进行社区发现,DeepWalk、M-NMF、DNGR、GAE、VGAE、GATE、AGRE[24]、AVGRE[24]、DEC、DAEGC、SDCN 等算法基于网络嵌入进行社区发现。采用准确率(accuracy, A_{CC})、标准互信息(normalized mutual information, N_{MI})、调整兰德系数(adjusted rand index, A_{RI})和 $F1$ 分数作为评价指标。试验数据集的基本信息如表 1 所示。

表 1 试验数据集基本信息

Table 1 Description of data sets

数据集	节点数量/个	边/条	社区数量/个	特征维度
Cora	2 708	5 429	7	1 433
ACM	3 025	13 128	3	1 870
Citeseer	3 327	4 732	6	3 703
DBLP	4 058	3 528	4	334

对比算法采用文献提供的默认参数,所提算法 DNECI 中社区数 K 为网络中的真实社区数,$\alpha=0.1$, $\beta=0.1$, $\gamma=0.05$, $T_1=100$, $T_2=50$, $T_3=50$,采用 Adam 优化器优化网络权重,学习率为 0.001。重启的随机游走步数设为 3。DNECI 采用 2 层图注意力神经网络,结构嵌入维度和属性嵌入维度分别为 16,融合嵌入维度为 32。

3.2 试验结果及分析

表 2 给出了 DNECI 与 3 种传统社区发现算法的试验对比结果,加粗数字表示最优值。可以看出,算法 DNECI 明显优于对比算法,DNECI 算法考虑了网络属性信息。这表明在真实的复杂网络中,不仅拓扑信息对社区结构有关联,网络的属性信息也发挥重要作用。

表 2 DNECI 和传统社区算法的对比试验结果

Table 2 Comparisons with traditional community algorithms

数据集	评价指标	对比试验指标			
		LPA	Infomap	BGLL	DNECI
Cora	N_{MI}	0.403	0.402	0.457	**0.534**
	A_{RI}	0.033	0.041	0.274	**0.514**
ACM	N_{MI}	0.197	0.243	0.182	**0.645**
	A_{RI}	0.008	0.012	0.009	**0.768**
Citeseer	N_{MI}	0.177	0.337	0.102	**0.378**
	A_{RI}	0.023	0.031	0.045	**0.393**
DBLP	N_{MI}	0.115	0.279	0.068	**0.399**
	A_{RI}	0.035	0.088	0.043	**0.431**

表 3 给出 DNECI 与其它 12 个利用节点嵌入进行社区发现算法的对比试验结果,加粗数字表示最优值。在 Cora 数据集上,DNECI 比排在第二的 DAEGC 算法在 A_{CC}、N_{MI} 和 A_{RI} 指标上提高了 2.4%、3.2%和 4.5%,在 $F1$ 分数上与 DAEGC 表现相当。在 A_{CM}数据集上,DNECI 的 A_{CC}、A_{RI} 和 $F1$ 等指标高于第二名 2.4%,3.5%,1.1%。在 Citeseer 数据集上,

DNECI 在 A_{CC}、A_{RI} 和 $F1$ 上均表现胜过其他算法。在 DBLP 数据集上,DNECI 的 A_{CC}、N_{MI}、A_{RI} 和 $F1$ 等指标分别比排名第二的算法提高了 3.4%、4.7%、4.3% 和 2.8%。

表 3　各算法在 4 个数据集上的指标值对比

Table 3　Index value comparison of different algorithms on 4 datasets

聚类算法	Cora				ACM				Citeseer				DBLP			
	A_{CC}	N_{MI}	A_{RI}	$F1$	A_{CC}	N_{MI}	A_{RI}	$F1$	A_{CC}	N_{MI}	A_{RI}	$F1$	A_{CC}	N_{MI}	A_{RI}	$F1$
k-means	0.417	0.233	0.140	0.439	0.681	0.328	0.315	0.684	0.445	0.214	0.186	0.416	0.396	0.119	0.074	0.320
DeepWalk	0.529	0.384	0.291	0.435	0.723	0.407	0.368	0.734	0.390	0.131	0.137	0.300	0.586	0.277	0.298	0.543
M-NMF	0.423	0.256	0.161	0.320	0.704	0.362	0.344	0.687	0.336	0.099	0.070	0.255	0.496	0.233	0.210	0.514
DNGR	0.419	0.318	0.142	0.356	0.718	0.384	0.326	0.695	0.326	0.180	0.043	0.286	0.534	0.265	0.256	0.558
GAE	0.627	0.496	0.418	0.611	0.863	0.592	0.641	0.864	0.581	0.327	0.308	0.575	0.623	0.325	0.238	0.631
VGAE	0.614	0.434	0.372	0.591	0.868	0.576	0.638	0.867	0.608	0.322	0.326	0.585	0.605	0.229	0.234	0.602
ARGE	0.640	0.449	0.352	0.587	0.852	0.589	0.654	0.871	0.610	0.351	0.350	0.584	0.618	0.262	0.218	0.620
ARVGE	0.638	0.450	0.374	0.377	0.846	0.590	0.647	0.858	0.615	0.347	0.353	0.575	0.620	0.257	0.231	0.632
DEC	0.565	0.376	0.327	0.583	0.827	0.507	0.551	0.829	0.563	0.282	0.277	0.535	0.573	0.285	0.213	0.586
GATE	0.664	0.462	0.425	0.640	0.871	0.596	0.658	0.868	0.634	0.362	0.370	0.594	0.638	0.268	0.284	0.629
DAEGC	0.699	0.517	0.492	**0.672**	0.877	0.619	0.672	0.877	0.647	**0.381**	0.384	0.603	0.653	0.273	0.302	0.645
SDCN	0.668	0.511	0.447	0.624	0.891	**0.654**	0.742	0.882	0.656	**0.381**	0.379	0.577	0.676	0.381	0.413	0.620
DNECI	**0.716**	**0.534**	**0.514**	**0.672**	**0.913**	0.645	**0.768**	**0.892**	**0.668**	0.378	**0.393**	**0.621**	**0.699**	**0.399**	**0.431**	**0.663**

进一步分析知,仅利用属性特征社区发现结果要优于只利用拓扑信息社区发现结果,真实网络数据中存在假阳性或假阴性连接,大规模复杂网络存在稀疏性问题,造成利用拓扑信息导致发现内部连接相对稀疏的社区表现受限。

基于网络嵌入进行社区发现要比直接进行社区发现表现更优,低维嵌入保留原始数据有效特征的同时,缓解了数据高维度和稀疏性影响。社区发现任务与网络嵌入耦合进行优化算法表现优于解耦优化算法,面向聚类任务网络嵌入更适合社区发现下游任务,可以提高社区发现的质量。

图 2 展示了在 ACM 数据集上节点拓扑相关性可视化图,颜色由暗到亮代表节点间相关性递增,亮色的点表示节点之间具有较高拓扑相关性。由于网络稀疏性,很多节点并没有存在相关性,通过随机游走,节点除了与直接邻居有拓扑连接,还可以与高阶邻域内的节点建立拓扑相关性,更充分挖掘网络本身的拓扑结构信息。

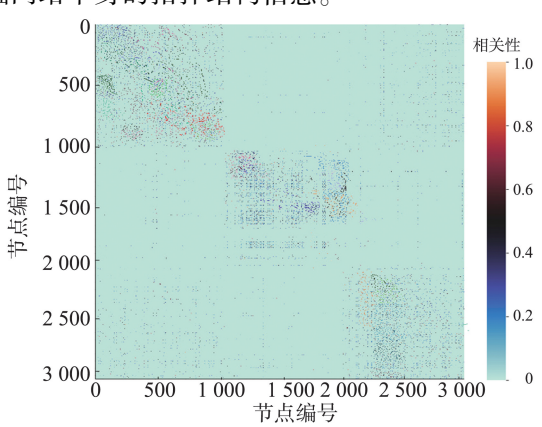


图 2　ACM 数据集上互相关性可视化图

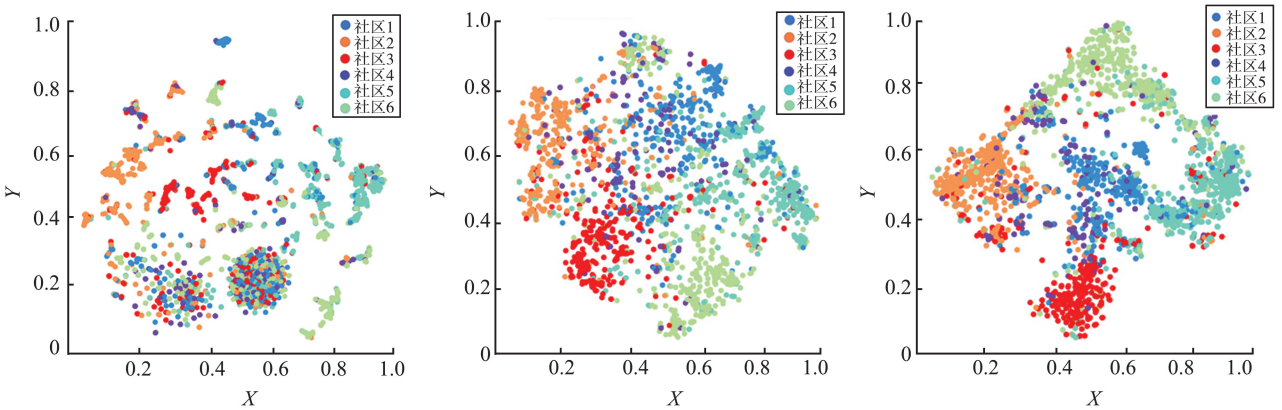
Fig.2　Visualization of cross-correlation on the ACM dataset

图 3 展示了在 Citeseer 数据集对拓扑视角嵌入、属性视角嵌入和融合嵌入的可视化效果图。由图 3(a)知,网络拓扑视角得到嵌入会出现不同社区节点混在一起的情况,这是由于挖掘网络高阶结构,使邻近但非同类的点被分配至同一社区,高阶领域内拓扑相关性节点分配至同一社区,造成同类的节点较为分散。由图 3(b)知,属性视角得到嵌入同一类的节点比较集中,类与类之间边缘比较模糊,类边缘节点会混淆在一起,这是由于图注意力网络进行嵌入时,利用了网络属性信息和节点低阶领域信息,低阶邻域作用使节点与节点较为集中,导致处于类边缘的点混在一起。由图 3(c)知,网络拓扑与节点属性融合得到的嵌入同类节点密集,不同类之间边缘清晰,类间距较大,符合社区“低耦合、高内聚”特点,融合嵌入同时考虑拓扑相关性与属性相关性,两种相关性强的节点被合理分配至同一社区,相关性不强的节点之间,网络高阶结构引入缓解了拓扑与属性不匹配的噪声影响。

图 4 展示在 Cora 数据集上运用不同聚类方法得到社区结果可视化图,其中图 4(a)为模块度优化社区划分结果,图 4(b)为聚类优化社区划分结果,图 4(c)为聚类集成社区划分结果。由图 4(a)(b)知,嵌入融合层对不同视角的网络嵌入进行了融合,在融合后的嵌入使用单独模块度优化或者聚类优化进行社区发现,可以得到较为理想的社区划分结果。图 4(b)中存在不同社区的边缘混淆情况。图 4(c)中可知,对两种社区发现方法进行集成优化,得到社区其内部节点分布更加紧密,

不同社区之间边缘清晰，较少出现混淆的情况，社区与社区间距离较远，更符合社区“低耦合、高内聚”特点。

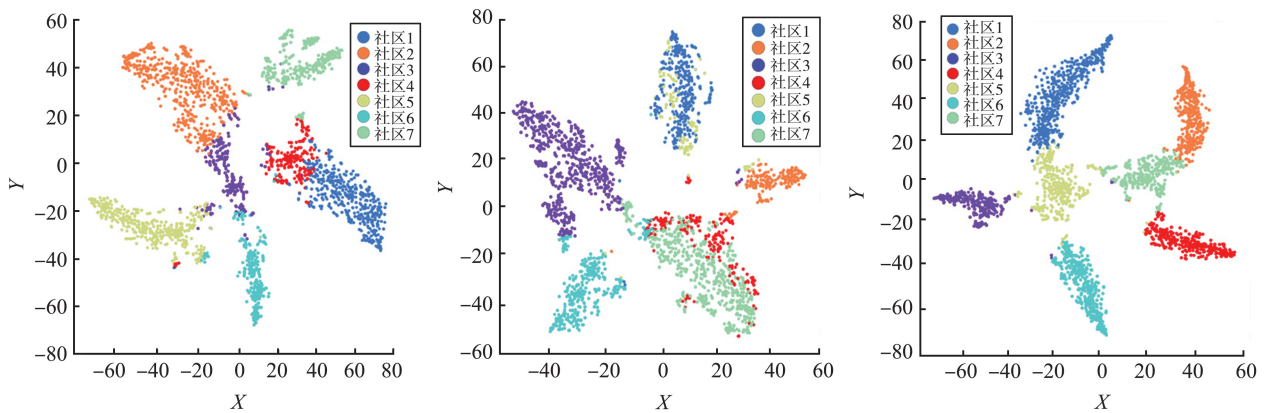
图 5 展示了 DBLP 数据集中不同 α、β 和 γ 对试验的影响，设置 α、β 为 $\{0.01,0.05,0.1,0.5,1,10\}$，$\gamma$ 为 $\{0.005,0.01,0.05,0.1,0.5,1\}$。结果表明，$\alpha$、$\beta$ 和 γ 的不同取值，会影响本研究算法试验结果。当 α、β 和 γ 分别取 0.1，0.1 和 0.05 时，本研究算法的性能较好，选择 $\alpha=0.1$，$\beta=0.1$，$\gamma=0.05$ 为本研究算法的参数。



(a) 拓扑视角嵌入可视化 (b) 属性视角嵌入可视化 (c) 融合视角嵌入可视化

图 3 Citeseer 数据集上不同嵌入可视化

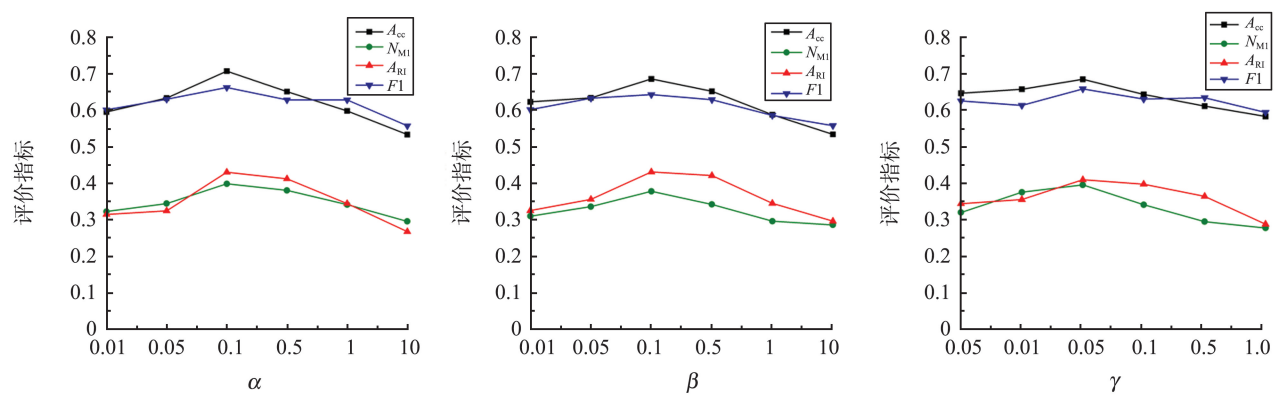
Fig.3 Visualization of the different perspective embedding on the Citeseer dataset



(a) 模块度优化社区发现结果可视化 (b) 聚类优化社区发现结果可视化 (c) 聚类集成社区发现结果可视化

图 4 Cora 数据集上不同聚类方法社区发现结果可视

Fig.4 Visualization of community discovery results ofdifferent clustering methods on the Cora dataset



(a) 不同α对试验影响 (b) 不同β对试验影响 (c) 不同γ对试验影响

图 5 DBLP 数据集上不同 α、β、γ 对试验影响

Fig.5 The impact of different α、β、γ values on the experiment in the DBLP dataset

为验证 DNECI 算法拓扑高阶结构与聚类集成模块各部分有效性,在不同数据集上进行消融试验,消融试验网络模块设置如表 4。不同数据集上消融试验对 A_{CC} 影响如表 5。由表 5 可知,DNECI_N_P 没有考虑高阶结构信息,试验结果低于其他引入拓扑高阶结构算法,表明属性网络中,拓扑高阶结构信息也起到重要作用。没有引入聚类集成,DNECI_N_Mod 与 DNECI_N_Clu 的 A_{CC} 会存在差异,引入聚类集成机制后,DNECI 与 DNECI_Mod 与结果较相近,社区发现结果更优,选取自监督聚类后社区发现结果作为社区结果。

表 4　消融试验模块设置

Table 4　Ablation study module settings

模块组合	PPMI 高阶结构	模块度优化	聚类优化	聚类集成
DNECI_N_P		√	√	√
DNECI_N_Clu	√	√		
DNECI_N_Mod	√		√	
DNECI_Mod	√	√	√	√
DNECI	√	√	√	√

注:DNECI_Mod 与 DNECI 包含各部分试验模块,DNECI_Mod 以模块度优化得到的社区结果作为最终的试验结果,DNECI 以聚类优化得到的社区结果作为最终的试验结果。

表 5　不同数据集上的 A_{CC} 消融试验结果

Table 5　Ablation study results of A_{CC} on different datasets

模块组合	消融试验 A_{CC} 结果			
	Cora	ACM	Citeseer	DBLP
DNECI_N_P	0.669	0.834	0.615	0.638
DNECI_N_Clu	0.643	0.877	0.634	0.653
DNECI_N_Mod	0.663	0.853	0.646	0.676
DNECI_Mod	0.702	0.902	0.659	0.687
DNECI	0.716	0.913	0.668	0.699

DNECI 算法将网络的属性视角和拓扑视角的信息有效融合,在提取网络的本身拓扑结构信息时,需要对包含网络拓扑相关性的 PPMI 进行嵌入处理,在优化时不需要额外代价计算 PPMI 及其嵌入,这可视作算法的数据处理的部分。通过邻接矩阵可计算得到模块度矩阵,在优化时不需要额外代价计算模块度矩阵,也可视作网络的数据处理部分。同时进行模块度优化和聚类优化,因此算法 DNECI 与 DSNE 等算法相比,没有增加计算代价的情况下提高了算法的性能,提高了社区的发现质量。

4　结论

本研究提出 DNECI 算法,由双视角网络嵌入和聚类集成两部分构成,双视角网络嵌入模块实现了网络属性信息与拓扑信息两种视角信息自适应融合,保留了网络属性信息与拓扑高阶结构。聚类集成模块中,模块度优化组件从拓扑结构上得到具有最优模块度的社区结果,聚类优化组件采用自监督聚类方法获取节点在嵌入空间的聚类结果,引入互监督机制将两种社区发现结果集成,保证了模块度优化和自监督聚类结果的一致。试验表明,DNECI 相比其他基准算法,有较好社区发现效果。本研究算法是基于无权无向的静态网络来进行的,在加权有向网络以及动态网络上进行社区发现未来值得探究的问题。

参考文献:

[1] BLONDEL V D, GUILLAUME J, LAMBIOTTE R. Fast Unfolding of Communities in Large Networks[J]. Journal of Statistical Mechanics: Theory and Experiment, 2008, 2008(10):10008.

[2] NEWMAN M E J. Fast algorithm for detecting community structure in networks[J]. Physical Review E, 2004, 69(6):66133.

[3] CLAUSET A, NEWMAN M E J, MOORE C. Finding community structure in very large networks[J]. Physical Review E, 2004, 70(2): 66111.

[4] RAGHAVAN U N, ALBERT R, KUMARA S. Near linear time algorithm to detect community structures in large-scale networks[J]. Physical Review E, 2007, 76(3): 36106.

[5] BARBER M J, CLARK J W. Detecting network communities by propagating labels under constraints[J]. Physical Review E, 2009, 80(2): 26129.

[6] BELKIN M, NIYOGI P. Laplacian eigenmaps and spectral techniques for embedding and clustering[J]. Advances in Neural Information Processing Systems, 2001, 14(6): 585-591.

[7] TENENBAUM J B, DE SILVA V, LANGGORD J C. A global geometric framework for nonlinear dimensionality reduction[J]. Science, 2000, 290(5500): 2319-2323.

[8] ROWEIS S T, SAUL L K. Nonlinear dimensionality reduction by locally linear embedding[J]. Science, 2000(290): 2323-2326.

[9] PEROZZI B, AL-RFOU R, SKIENA S. DeepWalk: online learning of social representations [C]// Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM, 2014: 701-710.

[10] GROVER A, LESKOVEC J. Node2vec: scalable feature learning for networks[C]// Proceedings of the 22th ACM SIGKDD International Conference on Knowledge

Discovery and Data Mining. New York, USA: ACM, 2016: 855-864.

[11] WANG D X, CUI P, ZHU WW. Structural deep network embedding[C]// Proceedings of 22th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM, 2016: 1225-1234.

[12] CAO SS, LU W, XU Q K. Deep neural networks for learning graph representations[C]// Proceedings of the 30th AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI Press, 2016: 1145-1152.

[13] WANG Xiao, CUI Peng, WANG Jing, et al. Community preserving network embedding [C]// Proceedings of the 32th AAAI Conference on Artificial Intelligence. Palo Alto, USA: AAAI press, 2017: 203-209.

[14] YANG C, LIU Z Y, ZHAO D L. Network representation learning with rich text information[C]// Proceedings of the 24th International Joint Conference on Artificial Intelligence. Palo Alto, USA: AAAI Press, 2015: 2111-2117.

[15] HUANG X, LI J D, HU X. Accelerated attributed network embedding [C]// Proceedings of the 2017 SIAM International Conference on Data Mining. Philadelphia, USA: SIAM, 2017: 633-641.

[16] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[C]// International Conference on Learning Representations. Toulon, France: ICLR, 2016: 718-725.

[17] VELICKOVIC P, CUCURULL G, GASANOVA A, et al. Graph attention networks[C]//Proceedings of the Int Conf on Learning Representations. Vancouver, Canada: ICLR, 2018: 485-497.

[18] KIPF T N, WELLING M. Variational graph auto-encoders[C]// Proceedings of the NIPS Workshopon Bayesian Deep Learning. Barcelona, Spain: NIPS, 2016: 1611-1616.

[19] SALEHI A, DAVULCU H. Graph attention auto-encoders [C]// Proceedings of the 32nd IEEE International Conference on Tools with Artificial Intelligence. Piscataway, USA: IEEE, 2020: 989-996.

[20] XIE J Y, GIRSHICK R, FARHADI A. Unsupervised deep embedding for clustering analysis [C]// Proceedings of the 33rd International Conference on Machine Learning. New York, USA: ACM, 2016: 478-487.

[21] WANG C, PAN S R, HU R Q, et al. Attributed graph clustering: a deep attentional embedding approach[C]// Proceedings of 28th International Joint Conference on Artificial Intelligence. San Francisco, USA: Morgan Kaufmann, 2019: 3670-3676.

[22] BO D Y, WANG X, SHI C, et al. Structural deep clustering network [C] // Proceedings of the 29th International World Wide Web Conference. New York, USA: ACM, 2020: 1400-1410.

[23] 郑文萍,王英楠,杨贵.基于双监督网络嵌入的社区发现算法[J]. 模式识别与人工智能, 2022, 35(3): 283-290.
ZHENG Wenping, WANG Yingnan, YANG Gui. Dual supervised network embedding based community detection algorithm [J]. Pattern Recognition and Artificial Intelligence, 2022, 35(3): 283-290.

[24] PAN Shirui, HU Ruiqi, JIANG Jing, et al. Adversarially regularized graph auto-encoder for graph embedding[C]// Proceedings of the 27th International Joint Conference on Artificial Intelligence. Melbourne, Australia: IJCAI, 2018: 2609-2615.

(编辑:陈燕)