研究与开发

基于高阶相似性的属性网络表示学习

邬少清，董一鸿，王雄，曹燕，辛宇
（宁波大学信息科学与工程学院，浙江 宁波 315211）

摘　要：现有的网络表示学习方法缺少对网络中隐含的深层次信息进行挖掘和利用。对网络中的潜在信息做进一步挖掘，提出了潜在的模式结构相似性，定义了网络结构间的相似度分数，用以衡量各个结构之间的相似性，使节点可以跨越不相干的顶点，获取全局结构上的高阶相似性。利用深度学习，融合多个信息源共同参与训练，弥补随机游走带来的不足，使得多个信息源信息之间紧密结合、互相补充，以达到最优的效果。实验选取 Lap、DeepWalk、TADW、SDNE、CANE 作为对比方法，将 3 个真实世界网络作为数据集来验证模型的有效性，进行节点分类和链路重构的实验。在节点分类中针对不同数据集和训练比例，性能平均提升 1.7 个百分点；链路重构实验中，仅需一半维度便实现了更好的性能，最后讨论了不同网络深度下模型的性能提升，通过增加模型的深度，节点分类的平均性能增加了 1.1 个百分点。

关键词：网络表示学习；图嵌入；属性网络；结构信息
中图分类号：TP311
文献标识码：A
doi: 10.11959/j.issn.1000−0801.2020309

Learning attribute network algorithm based on high-order similarity

WU Shaoqing, DONG Yihong, WANG Xiong, CAO Yan, XIN Yu
Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo 315211, China

Abstract: Due to the lack of deep-level information mining and utilization in the existing network representation learning methods, the potential pattern structure similarity was proposed by further exploring the potential information in the network. The similarity score between network structures was defined to measure the similarity between various structures so that nodes could cross irrelevant vertices to obtain high-order similarities on the global structure. In order to achieve the best effect, deep learning was used to fuse multiple information sources to participate in training together to make up for the deficiency of random walks. In the experiment, Lap, DeepWalk, TADW, SDNE and CANE were selected as comparison methods, and three real-world networks were used as data sets to verify the validity of the model, and experiments of node classification and link reconstruction are carried out. In the node classification, the average performance is improved by 1.7 percentage points for different datasets and training proportions.

收稿日期：2020−02−20；**修回日期**：2020−11−27
基金项目：浙江省自然科学基金资助项目（No.LY20F020009，No.LZ20F020001）；国家自然科学基金资助项目（No.61602133）；宁波市自然科学基金资助项目（No.202003N4086，No.2019A610093）
Foundation Items: Zhejiang Provincial Natural Science Foundation of China (No.LY20F020009, No.LZ20F020001), The National Natural Science Foundation of China (No.61602133), Ningbo Natural Science Foundation (No.202003N4086, No.2019A610093)

In the link reconstruction experiment, only half the dimension is needed to achieve better performance. Finally, the performance improvement of the model under different network depths was discussed. By increasing the depth of the model, the average performance of node classification increased by 1.1 percentage points.

Key words: network representation learning, graph embedding, attribute network, structure information

1　引言

在现实世界中网络无处不在，如引文网络[1]、蛋白质网络[2]、社交网络[3]，网络数据的复杂性给机器学习任务带来了巨大挑战，这些算法需要一组具有信息性、区分性和独立性的特征。一种有效的方法是将网络嵌入一个共同的、连续的低维空间中，其目标是在学习的嵌入空间中重构网络。旨在学习网络低维矢量表示的网络嵌入（网络表示学习）已被证明在许多网络分析任务中非常有效，如节点分类[4]、推荐[5]和链接预测[6]。已经提出多种方法来实现这一目标，通常包括 DeepWalk[7]、LINE[8]、GraRep[9]和 node2vec[10]。这些模型在多个现实世界的网络中被证明是有效的。

最早的网络表示方法基于随机游走，利用节点之间的链路连接，该方法存在的一个限制是如果结构相似的节点距离（跳数）大于 Skip-Gram[11] 窗口，那么它们永远不会共享相同的上下文，学到的节点向量只能表示节点之间的低阶局部相似性。为获取网络节点之间的全局相似性，常用的方法是利用节点附带的属性信息。属性信息的相似性，不依赖于局部的链路连接，因此可以获得全局层面上的相似性。

节点属性所带来的相似性是显式可见的信息，本文的工作拟挖掘并利用网络的潜在信息来构建节点之间的高阶相似性，以获得更好的节点表示。真实网络中存在着众多分布较远却有相同模式特征的结构，拥有相似模式特征的节点并不一定相邻，甚至可能相距很远，若能跨越不相干顶点，加强它们的联系，就能更好地捕获节点在全局层面上的相似性。并且每个结构内部节点通常密集地连接在一起[12]，共享某些共同的属性，这也有利于网络分析任务。

针对先前网络表示学习方法无法利用网络潜在信息的缺陷，提出了基于高阶相似性的属性网络表示学习算法（learning attribute network algorithm based on high-order similarity，LANAHS）的模型，本文主要贡献如下。

- 对网络中的潜在信息做进一步挖掘，提出了潜在模式结构相似性，定义了网络结构间的相似度分数，使节点可以跨越不相干的顶点，加强在全局层面上的联系。
- 利用属性相似性和模式结构相似性弥补随机游走带来的不足，将 3 方面的信息做早期融合后，送入深度神经网络参与训练，克服了网络中不同信息各自学习带来的信息丢失。
- 在 3 个真实网络和两个网络分析任务上对 LANAHS 及其变体进行了广泛的评价，证明了模型的有效性。

2　相关工作

近些年来，随着大规模社交网络的发展，网络表示学习被提出以解决大规模网络的分析任务，该技术有助于针对复杂网络数据的下游机器学习的应用，因为节点嵌入可以直接用于分类和链路预测等任务。

DeepWalk 首先提出了从网络中学习语言模型，通过随机游走产生一系列顶点序列后使用 Skip-Gram 模型学习顶点嵌入。受 DeepWalk 的启发，node2vec 探索不同的随机游走策略，捕获顶点的同质性和结构等同性。Line 分别保留了网络的一阶和二阶接近度。针对浅层模型无法捕获高度非线性的网络结构的缺点，SDNE[13]使用深层神

经网络对节点表示间的非线性进行建模。NECS[14]使用非负矩阵分解学习顶点表示，并根据同构原理对社区结构进行建模，结合社区指导顶点表示学习。然而，这些方法仅考虑节点的链路连接信息，而忽略了真实网络中丰富的节点属性信息。

为充分利用属性信息，研究者提出了一系列工作来融合网络中的节点属性信息。TADW[15]通过分解与文本相关的矩阵来合并顶点的文本特征。AANE[16]提出分布式算法，将整个优化过程分解为 $2n$ 个简单的独立子问题。CANE[17]通过考虑网络节点附加的文本信息和引入相互注意力机制，学习上下文相关的节点表示。NEEC[18]在表示学习中引入了外部的专家认知，将专家知识建模为新边缘。RoSANE[19]将网络拓扑结构和节点属性集成在一起构建密集网络，克服网络稀疏性。NetVAE[20]是为网络嵌入设置的变分编码器，分别对链路结构和属性信息设计不同解码器。Net2Net-NE[21]通过自我中心网络的中心节点与边节点之间的强相关性，将结构和属性信息融合到中心节点嵌入中。

3 基本概念

定义 1 属性网络[17]

属性网络定义为 $G=(V,E,T)$，其中 $V=\{v_1,v_2,\cdots,v_{|V|}\}$ 是顶点集合，$E\subseteq(V\times V)=\{e_{ij}\}$ 是边集合，每个边缘 e_{ij} 与权重 s_{ij} 相关联，表示 v_i 和 v_j 之间的连接强度，本文仅关注未加权网络。T 表示节点的文本内容，$T=\{(v,\mathrm{doc})\,|\,v\in V;\mathrm{doc}=\{t_i\}\}$，其中 t_i 表示 doc 的第 i 个句子，由单词序列 $<w_1,w_2,\cdots,w_m>$ 组成。在不失一般性的情况下，假设网络结构是有向图。

定义 2 网络嵌入[17]

给定一个表示为 $G=(V,E,T)$ 的属性网络，网络嵌入的目的是为每个顶点 $v\in V$ 分配一个低维实值矢量，表示为 $v\in\mathbb{R}^d$，其中 $d\ll|V|$，该表示向量蕴含了节点的网络链路信息和其他附加信息。可以视为顶点 v 的特征向量，用作后续任务的输入。

4 模型框架

首先介绍如何通过深度神经网络对链路结构建模，然后介绍高阶模式结构相似性的建模，对模式结构进行表征，之后介绍了属性相似性的建模。最后介绍如何综合三方信息源进行联合训练，并提供了最终的目标函数。

4.1 链路结构建模

网络链路结构建模由构建语料库 S 开始，从每个节点开始多次截断随机游走，获取一系列的游走序列。目标是使出现在一个窗口中的节点的共现概率最大化，即在给定当前节点 v_i 的情况下，最大化该节点到周围节点的概率，则在单一游走序列中的损失函数如式（1）所示：

$$\mathcal{L}=\sum_{t=1}^{T}\log\mathbb{P}\left(v_{t-b}:v_{t+b}|v_t\right) \tag{1}$$

其中，b 是窗口大小，$v_{t-b}:v_{t+b}$ 是不包括节点 v_t 的节点序列。假设窗口内节点彼此独立，则概率 $\mathbb{P}\left(v_{t-b}:v_{t+b}|v_t\right)$ 如式（2）所示：

$$\prod_{-b\leqslant j\leqslant b,j\neq 0}\mathbb{P}\left(v_{t+j}|v_t\right) \tag{2}$$

通过最大化给定节点到其邻域内所有节点的条件概率，可以保持链路结构上的接近度。将式（2）拓展到所有游走序列，全局链路结构建模的似然函数如式（3）所示：

$$\begin{aligned}\mathcal{L}=&\sum_{i=1}^{N}\sum_{s\in S}\log\mathbb{P}\left(v_{i-b}:v_{i+b}|v_i\right)=\\&\sum_{i=1}^{N}\sum_{s\in S}\sum_{-b\leqslant j\leqslant b,j\neq 0}\log\mathbb{P}\left(v_{i+j}|v_i\right)\end{aligned} \tag{3}$$

其中，$\mathbb{P}\left(v_{i+j}|v_i\right)$ 表示采样节点对之间接近度的得分，使用 softmax 函数，如式（4）所示：

$$\mathbb{P}\left(v_{i+j}|v_i\right)=\frac{\exp\left(f\left(v_{i+j},v_i\right)\right)}{\sum_{j'=1}^{N}\exp\left(f\left(v_i,u_{j'}\right)\right)} \tag{4}$$

式（4）用来测量节点 v_i 和 v_{i+j} 之间连接的可能性，f 为映射函数，用于将两个节点映射到它们的成对接近度，此前工作多使用浅层模型来进行关系建模，在浅层模型中，两个节点的接近度通常被建模为嵌入向量的内积，但浅层模型无法捕获高度非线性的网络结构，从而导致次优的网络表示。为了捕捉现实世界网络的复杂非线性，采用深度架构来模拟节点的成对接近度，如式（5）所示：

$$f\left(v_i, v_{i+j}\right) = V'_{v_{i+j}} \delta_n\left(\boldsymbol{W}^{(n)}\left(\cdots \delta_1\left(\boldsymbol{W}^{(1)} V_{v_i} + \boldsymbol{b}^{(1)}\right)\cdots\right) + \boldsymbol{b}^{(n)}\right) \tag{5}$$

其中，$\boldsymbol{V}_{v_i}$ 表示节点 v_i 的嵌入向量，$\boldsymbol{V}'_{v_{i+j}}$ 表示将节点 v_{i+j} 嵌入为邻居时的向量，n 表示将嵌入向量转换为最终表示的隐藏层数量，$\boldsymbol{W}^{(n)}$、$\boldsymbol{b}^{(n)}$ 和 δ_n 分别表示第 n 个隐藏层的权重矩阵、偏置向量和激活函数。

4.2　高阶相似性建模

高阶相似性通常指超越一阶直接相连和二阶邻域结构之外的相似性[8-9]，旨在使节点摆脱局部链路限制，探索节点更高层面上的全局相似性。本文通过模式结构来探究节点之间的高阶相似性，模式结构相似性可使节点跨越不相干顶点，加强彼此联系，超越一阶二阶限制，体现出高阶相似性。在学习过程中，融入模式结构相似性可以使最终得到的嵌入向量表示更加准确，提升下游应用的性能。

真实网络中存在众多分布较远却有着相同模式特征的结构（子图），子图结构的形状通过构造“词汇表”预先定义，内含真实图形中经常出现的结构。参考网络拓扑结构，将基本结构定义为链式、团状、星形和二分图 4 种类型。这些“词汇”术语在真实网络中经常出现，并且具有具体语义含义。图 1 分别描述了各个基本结构的特征。

- 链式（chain）结构，存在起始节点和尾节点，位于二者之间的节点以线形方式依次相连。
- 星形（star）结构，存在一个中心节点和 n 个与中心节点相连但彼此互不相连的轮辐节点，轮辐顶点之间互相等价。
- 团状（clique）结构，满足两两之间有边连接的顶点集合，所有顶点完全等价。
- 二分图（bipartite graph），顶点可以分成两个互斥的独立集 A 和 B，所有边都是连结一个 A 中的点和一个 B 中的点。

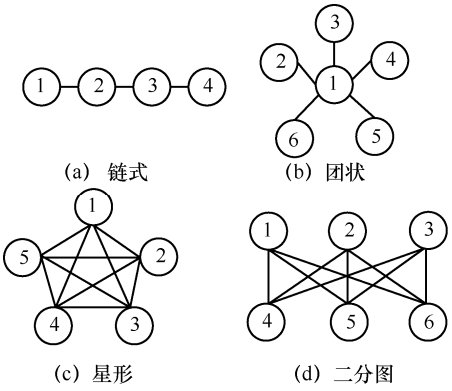


图 1　4 种基本结构

网络中并非所有子图结构都能被严格定义，更多的结构只是类似于基本结构，称为近似结构。如图 2 所示，以下 4 个子图都非基本结构，但都近似于某个或多个基本结构，例如图 2（a）所示的结构，不属于任何基本结构，但在直观上类似于链式结构或星形结构；图 2（b）所示的结构直观上类似于星形结构。

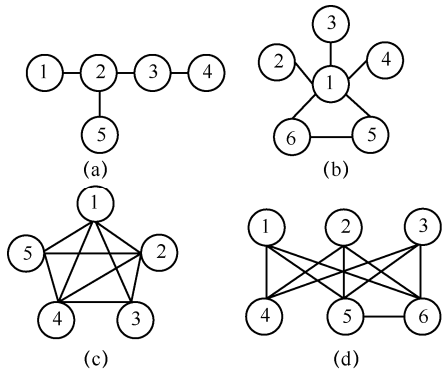


图 2　近似结构

如图 2 所示，图 2（b）近似星形结构转换为

图 1（b）星形结构，只需去除节点 5 和 6 之间的冗余边；图 2（c）近似团结构转换为图 1（c）团结构，只需在节点 3 和节点 5 添加一条边。由此可知近似结构和基本结构之间，通常存在着边的冗余或缺失，要完成近似结构到基本结构之间的转换，只需少量的编辑行为。

要定量地评价子图和各个基本结构之间的相似度，需要为每个结构生成一个相似度分数，用于衡量子图到各个基本结构之间的距离。因此给出各个近似结构下的编码规则，遵循最小描述长度（MDL）原则进行结构代价损失，在编码规则下计算所需的最小编码代价比，代价比越小，就越近似于该基本结构。

定义 3 相似度分数

给定子图的邻接矩阵和预先定义的基本结构集，子图 c 在基本结构模型 $\Omega=\{\text{ch},\text{st},\text{cl},\text{bc}\}$ 下的相似度分数定义为：

$$\text{Score}(c)=\frac{L(\Omega_i)-L(E)}{L(\Omega_i)} \tag{6}$$

$L(\Omega_i)$ 为描述结构的字节长度，$L(E)$为描述编辑行为的字节长度。其中，基本结构的质量分数为 1，此时 $L(E)$为 0，各个近似结构下质量分数的计算方式如下。

（1）将子图编码为链式结构，其结构编码为 $L(\text{ch})=\log(|\text{ch}|-1)+\sum_{i=0}^{|\text{ch}|}\log(n-i)$，$L(E)=\log(|A_V|-|\Pi_V|)+1$，先编码链中的节点数，然后按顺序编码它们的 ID，误差编码时无须考虑缺少边，而是新增边。$|\Pi_V|$表示简单结构状态下应存在的边数，$|A_V|$表示实际存在的边数。

（2）将子图编码为星形结构，其结构编码为 $L(\text{st})=\log(|\text{st}|-1)+\log n+\log C_{n-1}^{|\text{st}|-1}$，$L(E)=\log(|\Pi_V|-|A_V|)+1$，其中 C 表示组合运算，$|\text{st}|-1$ 是辐条的数量。意为从 n 个节点中识别出集线器（中心），从剩余节点中识别出辐条。

（3）将子图编码为团结构，其结构编码为 $L(\text{cl})=\log(|\text{cl}|)+\log C_n^{|\text{cl}|}$，$L(E)=\log(|\Pi_V|-|A_V|)+1$，首先编码节点的数量，然后编码节点 ID。

（4）将子图编码为二分图结构，$L(\text{bc})=\log(|A|)+\log(|B|)+\log C_n^{|A|+|B|}$，$L(E)=\log(|\Pi_V|-|A_V|)+1$，二分图被分为 A、B 两部分，首先对 A、B 的进行编码，然后编码节点 ID。

通过上述编码规则可得到子结构与各基本结构的相似度分数，一组这样的相似度分数构成一个相似度向量，该向量表示子结构与各个基本结构的相似性，该向量既是子结构的表征，也是子结构中每个节点的高阶特征。由于子图是重叠子图，一个节点可能同时隶属于多个子结构，因此选择最接近基本结构的子图作为节点所处的子结构，表征节点所处子图的结构相似性。根据定义 3，子图 c 与各个基本结构模型 Ω_i 的相似度分数为 $\text{Score}(c)_{\Omega_i}$，若节点隶属于多个子图结构，分别针对所属子结构计算 $\max\{\text{Score}(c)_{\Omega_i}\}$，选取拥有最佳相似度的子结构作为节点的唯一隶属结构。

为最大限度地利用挖掘出来的信息，本文还对每个节点所处的具体子图编号进行二进制编码，因为结构内部节点通常密集连接在一起[12]，共享某些属性，表明节点是否处于同一子图，将有利于节点的表示学习。最终获取的模式结构相似性向量，是节点的结构相似度向量和所属子图的编号向量的连接，降维后可得到模式结构相似性向量 S，作为高阶相似性的一部分参与联合训练。单就模式结构相似性而言，与链路结构建模类似，目标是通过采用深度模型来模拟节点所属模式之间相似性，可通过类似于式（5）的方式实现，使用 $\boldsymbol{s}_i$ 替换 $\boldsymbol{V}_{v_i}$，替换后如式（7）所示。

$$f(v_i,v_j)=s'_j\,\delta_n\left(\boldsymbol{W}^{(n)}\left(\cdots\delta_1\left(\boldsymbol{W}^{(1)}s_i+\boldsymbol{b}^{(1)}\right)\cdots\right)+\boldsymbol{b}^{(n)}\right) \tag{7}$$

其中，$\boldsymbol{s}_i$ 表示节点 v_i 的模式结构相似性向量，$\boldsymbol{s}'_j$ 表示将节点 v_j 嵌入为邻居时的模式结构相似性向

量，n 表示将嵌入向量转换为最终表示的隐藏层数量；$\boldsymbol{W}^{(n)}$、$\boldsymbol{b}^{(n)}$ 和 δ_n 分别表示第 n 个隐藏层的权重矩阵、偏置向量和激活函数。具体算法流程如下所示。

算法 1　模式结构相似性向量生成

输入　图 G，基本结构集 Ω。

输出　相似度分数，子图编号，子图候选集 $\boldsymbol{M}$，模式结构相似性 S。

使用图分解方法生成子图集 $\mathcal{C}$

for item in $\mathcal{C}$

if item $\in \Omega$

标记为基本结构，添加到子图候选集 M

else

搜索基本结构集 Ω，选择编码成本最低的结构，添加到子图候选集 M

for item in M

进行结构代价损失，计算相似度分数，合成相似性向量

for v_i in G

if $v_i \in$ 唯一子图 c

$s_i = c$相似性向量 $+ c$子图编号

else $v_i \in$ 多个子图 c

选取具有最大相似度的结构 c，$c \in \max\limits_{c}\{\text{Score}(c)_{\Omega}\}$

$s_i = c$相似性向量 $+ c$子图编号

return S

4.3　属性相似性建模

真实网络通常包含丰富的提供了额外信息的文本属性，并且属性信息之间的相似性不被局部链路结构限制，可以较好地体现节点在全局层面的相似性。Doc2vec[22]是一个不受约束的框架，用于学习文本片段的连续分布式向量表示，克服了基于统计的模型中没有语义的缺点。

Doc2vec 框架如图 3 所示，每个段落都映射到唯一的向量，由矩阵 $\boldsymbol{D}$ 中的列表示，每个单词也映射到唯一的向量，由矩阵 W 中的列表示。段落向量和单词向量被平均或连接以预测上下文中的下一个单词。段落标记可以充当记忆，记住当前上下文中缺少的内容或段落的主题，并在同一段落生成的所有上下文中共享。经过训练后，段落向量可以用作段落的特征，表征段落的语义。

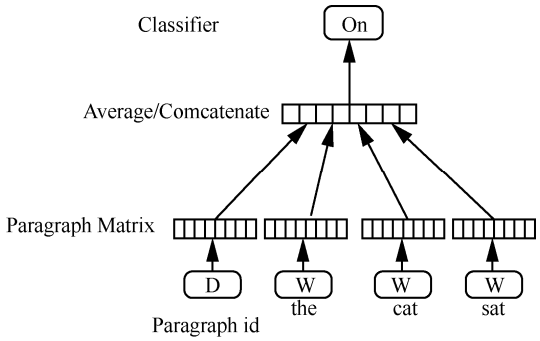


图 3　段落向量框架

通过训练，可以获得每一个节点属性的向量表示 $\boldsymbol{a}_i$，并参与联合训练。与链路结构建模类似，目标是通过采用深度模型近似属性之间的复杂相互作用来模拟属性邻近度，这可以通过类似式（5）和式（7）的方式来实现，同时用 $\boldsymbol{a}_i$ 替换 $\boldsymbol{V}_{v_i}$。

4.4　联合训练

利用高阶相似性和属性相似性最简洁的方法是做简单拼接，或者分别学习后做一个后期的融合降维。后期融合的主要缺点是各个模型在不知道彼此的情况下被单独训练，并且在训练之后简单地组合结果。一个较佳的解决方案是进行早期融合，使用高阶相似性和属性相似性对链路结构的学习进行时时补充。因此，提出了一个通用的属性网络嵌入框架，如图 4 所示。

模型采用多层神经网络，利用了深度学习强大的表示和泛化能力，通过输入层的早期融合来集成链路结构建模、属性相似性建模、模式结构相似性建模，将每部分得到的向量表示进行早期融合，送入多层非线性层中，模拟节点的成对接近度，节点对的选择通过随机游走方式获得，

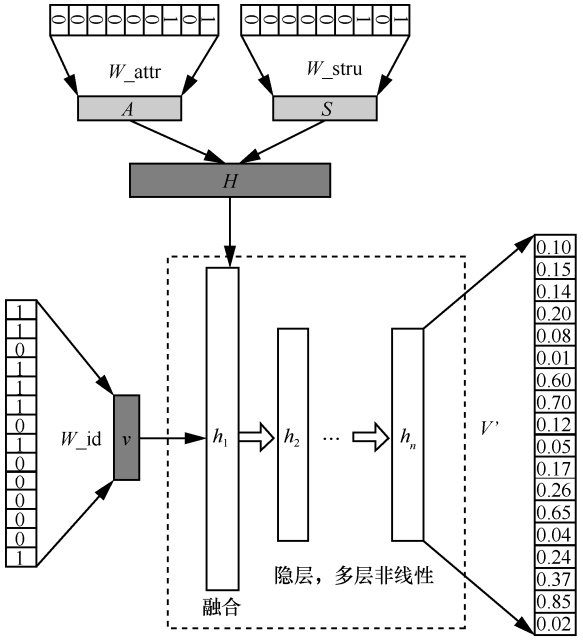


图 4　模型结构

通过从每个节点开始固定长度的游走，采样游走路径窗口内的节点对。最后将非线性层的输出向量转换为概率向量 $\boldsymbol{O}=\left[p(v_1|v_i),p(v_2|v_i),\cdots,p(v_N|v_i)\right]$，目标是最大化所有节点上的条件链接概率。具体设置如下。

融合层 $\boldsymbol{h}^{(1)}$ 聚合通过不同信息源得到的表示，即：

$$\boldsymbol{h}^{(1)}=\begin{bmatrix} v \\ \lambda \boldsymbol{H} \end{bmatrix} \tag{8}$$

其中，$\boldsymbol{H}$ 由属性表示 A 和模式结构相似性表示 S 拼接而成，共同对链路结构建模进行补充和纠正。$\lambda \in \mathbb{R}$ 用来调节高阶相似性的重要性。

$$\boldsymbol{h}^{(n)}=\delta_n\left(\boldsymbol{W}^{(n)}\boldsymbol{h}^{(n-1)}+\boldsymbol{b}^{(n)}\right), n=1,2,\cdots,l \tag{9}$$

其中，δ_n 表示激活函数，l 是隐藏层的数量。

最后需要将隐藏层 $\boldsymbol{h}_i^{(n)}$ 的输出向量转换为概率向量 $\boldsymbol{O}$，其包含 v_i 对其他所有节点的预测链路概率：

$$\boldsymbol{O}=\left[p(v_1|v_i),p(v_2|v_i),\cdots,p(v_N|v_i)\right] \tag{10}$$

依照前文可知：

$$p(v_j|v_i)=\frac{\exp(f(v_i,v_j))}{\sum_{j'=1}^{N}\exp(f(v_i,v_{j'}))} \tag{11}$$

其中，$f(v_i,v_j)=\boldsymbol{V}'_{v_j}\boldsymbol{h}_i^{(n)}$，$\boldsymbol{V}'_{v_j}$ 表示邻居 v_i 的表示，对应于最后隐藏层和输出层之间的权重矩阵 $\boldsymbol{V}'$ 中的行。具体算法流程如下所示。

算法 2　联合训练 LANAHS

输入　属性图 $G=(V,E,T)$，窗口大小 b，嵌入维度 d，游走步长 t，节点遍历次数 γ

输出　节点表示矩阵 $\boldsymbol{V}\in\mathbb{R}^{|V|\times d}$

初始化　从 $\boldsymbol{V}^{|V|\times d}$ 中采样 $\boldsymbol{V}$

for i=0 to γ do

$\mathcal{O}$=shuffle(V)

for each $v_i\in\mathcal{O}$

从v_i开始的游走的序列S_{v_i}=随机游走算法(G,v_i,t)

多层神经网络算法(Doc2vec+算法 1)

end for

end for

4.5　模型优化

为了估计整个框架的模型参数，需要指定一个优化的目标函数，如式（12）所示。目标是最大化所有节点上的条件链接概率。利用该目标函数，整个框架可以被联合训练，使得在该参数下最大化似然概率。

$$\begin{aligned}\theta^*&=\operatorname*{argmax}_{\theta}\prod_{i=1}^{N}\prod_{j\in N_i}p(v_j|v_i)=\\ &\operatorname*{argmax}_{\theta}\sum_{v_i\in N}\sum_{v_j\in N_i}\log p(v_j|v_i)=\\ &\operatorname*{argmax}_{\theta}\sum_{v_i\in N}\sum_{v_j\in N_i}\log\frac{\exp(f(v_i,v_j))}{\sum_{j'=1}^{N}\exp(f(v_i,v_{j'}))}\end{aligned} \tag{12}$$

式（12）中 softmax 方案最大化有两个效果，一是增强 v_i 和 $v\in\mathcal{N}_i$ 之间的相似性；二是减弱 v_i 和 $v\notin\mathcal{N}_i$ 之间的相似性。并且为了计算单个概率，需要遍历整个网络中的所有参与者，这在计算上是低效的。为了避免这些问题，应用负采样程序，只从整个社交网络中抽取非常小的用户子集。经过适当的优化后，得到节点的表示 $\boldsymbol{h}^{(n)}$ 和 $\boldsymbol{V}'$，使

用 $\boldsymbol{h}^{(n)}+\boldsymbol{V}'$ 作为每个节点的最终表现形式。

5　实验与分析

为验证模型对于节点之间关系建模的有效性，在多个真实数据集上进行节点分类和链路重构实验，本文方法和对比方法均在统一环境中运行，使用 Ubuntu 服务器，英伟达 2080TI 的 GPU，英特尔酷睿 i7-9700k 的 CPU，内存为 16 GB DDR4。

5.1　数据集

选取了如下 3 个来源于真实网络的数据集进行实验。

Cora[23]是学术论文引用网络，包含 2 277 篇与机器学习相关的论文，论文之间的相互引用次数为 5 214 次，即有 2 277 个节点、5 214 条边，根据研究方向的不同这些论文被分成 7 类。

DBLP[24]是由计算机科学书目组成的数据集。根据研究领域的不同，这些论文被分成 4 类。经过预处理后，网络由 13 404 个节点和 39 861 条边组成。

Facebook[25]是基于 Facebook 构建的用户社交网络数据集，用户节点来自两所美国大学的学生，节点属性为用户的个人基本资料。共有 18 163 个用户和 766 800 条边。

5.2　对比算法

使用以下算法与本文方法进行比较。

Lap[26]：Laplace 特征表早期基于矩阵特征向量计算的方法，优化问题类似地转化为 Laplace 矩阵的特征向量计算问题。

DeepWalk[7]：在网络上进行随机游走，生成节点的随机游走序列，采用训练词向量的 Skip-Gram 模型，学习网络节点表示。

TADW[15]：证明了 DeepWalk 实际上等同于矩阵分解，TADW 在矩阵分解框架下将顶点的文本特征结合到网络表示学习中。

SDNE[13]：首次将典型深层神经网络引入网络表示学习，通过引入深层自动编码器对节点的邻接向量进行编码，得到节点的低维实值表示向量。

CANE[17]：是与上下文相关的网络表示学习模型，通过考虑节点的文本信息和引入相互注意力机制，对节点之间的关系进行更准确的建模，学习上下文相关的节点表示。

LANAHS_A：是 LANAHS 的变体，和 LANAHS 采样相同的架构，但只考虑链路结构相似性和属性相似性，该方法用以验证模式结构相似性的有效性。

5.3　评价指标

节点分类：对于节点分类任务，顶点表示由网络嵌入方法生成，整个网络都用于学习网络表示。随机抽取标记顶点的一部分（30%~90%）作为训练数据，其余顶点用于测试。使用 scikit-learn[27]来训练逻辑回归分类器。对于每组训练数据，实验独立执行 10 次，报告平均准确率，准确率的计算式为：$\text{accuracy}=\dfrac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{FN}+\text{TN}}$。选择线性分类器，而不是非线性模型或复杂的关系分类器[28]，可减少复杂学习方法对分类性能的影响。

链路重构：对于链接重构任务，采用一个标准的评测指标 AUC[29]。该指标表示实际存在链接的两个节点之间的相似度大于不存在链接的两个节点的相似度的概率，计算式为 $\text{AUC}=\dfrac{\sum_{i\in\text{positiveClass}}\text{rank}_i-\dfrac{M(1+M)}{2}}{M\times N}$。实验中，随机抽取一半原始存在链路，将其标记为负样本，即不存在的链路，进行链路重构实验。

5.4　参数设定

本文采用随机游走采样网络节点，窗口大小设置为 10，步长设置为 40，顶点步数设置为 80。隐藏层激活函数选择 soft sign 函数，隐藏层数量默认为 1。使用高斯分布随机初始化模型参数，通过网格搜索调整批量大小和学习率，$\lambda=1$。为公

平比较，不同对比方法应保持最终嵌入维度的一致，参照参考文献[7-10]中的维度设置方法，节点分类中将所有向量维度设置为 128；链路重构中，为探究模型能否使用较少维度实现较高性能，在 16~128 维间选取 8 个维度进行比较。对比实验的其他参数依照参考文献中的最佳参数设置。

5.5 算法比较

选择 Cora 数据集和 DBLP 数据集进行节点分类实验，实验结果见表 1 和表 2。选择 Cora 数据集、DBLP 数据集和 Facebook 数据集进行链路重构实验，实验结果见表 3~表 5。

5.5.1 节点分类

在 Cora 数据集上，将网络上的训练比例从 30%提高到 90%进行节点分类任务，结果见表 1，粗体数字表示每列中的最高性能。由表 1 可知，本文的方法始终优其他先进模型。另外，LANAHS_A 在同样结构下，仅考虑属性和链路结构，而忽略模式结构，用以验证模式结构相似性的有效性，LANAHS_A 在所有情况下，性能都次于 LANAHS，但高于其他方法，证明了模式结构相似性的有效性，同时也证明了本文框架的有效性。在 DBLP 上进行同样的操作，本文所提出的方法依然优于其他对比方法，见表 2。

5.5.2 链路重构

在 Cora、DBLP 和 Facebook 数据集上，维度为 16～128 维，步长为 16，进行链路重构实验，粗体数字表示每行中的最高性能。由表 3 可知，本文提出的方法在不同维度下都达到了最佳性能。特别地，本文提出的方法在仅需 48 维的情况下，性能已经超过其他方法在 128 维的性能，显示了本文方法的卓越性能，只需不到一半的维度，就可以超越其他方法的最佳性能，表明本文方法具有更强的表现力。在大规模网络图上，减小存储量对于提升性能有较大帮助。

表 1 Cora 数据集节点分类实验结果

方法	30%	40%	50%	60%	70%	80%	90%
Lap	29.66%	30.54%	31.67%	33.87%	36.29%	39.53%	42.71%
DeepWalk	79.58%	80.88%	81.54%	82.12%	82.90%	82.83%	83.88%
SDNE	59.51%	61.74%	63.39%	64.36%	65.05%	65.61%	65.55%
TADW	53.49%	56.17%	57.92%	59.47%	60.29%	60.86%	61.72%
CANE	57.28%	60.25%	60.89%	63.27%	64.08%	64.65%	65.24%
LANAHS_A	81.02%	81.27%	82.26%	83.02%	83.71%	84.12%	84.31%
LANAHS	82.15%	82.84%	83.57%	84.21%	84.62%	85.06%	85.73%

表 2 DBLP 数据集节点分类实验结果

方法	30%	40%	50%	60%	70%	80%	90%
Lap	51.61%	52.86%	55.11%	56.64%	58.27%	59.71%	59.59%
DeepWalk	82.56%	82.71%	82.64%	82.65%	82.79%	83.17%	83.25%
SDNE	53.08%	53.10%	53.18%	53.26%	53.71%	54.08%	54.15%
TADW	82.50%	82.71%	82.91%	83.29%	83.29%	84.04%	84.04%
CANE	82.61%	82.82%	83.08%	83.12%	83.13%	83.24%	83.56%
LANAHS_A	83.01%	83.24%	83.59%	83.62%	83.73%	84.25%	84.25%
LANAHS	83.75%	84.75%	84.89%	85.26%	85.28%	85.43%	85.72%

表 3　Cora 数据集链路重构实验结果

方法	Lap	DeepWalk	SDNE	TADW	CANE	LANAHS_A	LANAHS
16 维	0.959 2	0.995 2	0.652 4	0.569 5	0.602 5	0.985 4	0.996 3
32 维	0.973 1	0.997 3	0.662 0	0.599 6	0.637 1	0.994 1	0.998 5
48 维	0.978 6	0.998 1	0.678 9	0.626 0	0.662 1	0.998 4	0.999 0
64 维	0.981 6	0.998 4	0.679 4	0.630 6	0.673 6	0.999 0	0.999 0
80 维	0.983 6	0.998 5	0.718 9	0.661 6	0.692 1	0.999 0	0.999 0
96 维	0.985 8	0.998 5	0.738 1	0.661 8	0.708 2	0.998 9	0.999 0
112 维	0.986 9	0.998 5	0.773 6	0.669 5	0.725 4	0.998 8	0.999 0
128 维	0.988 8	0.998 5	0.887 7	0.679 4	0.736 5	0.998 8	0.999 0

表 4　DBLP 数据集链路重构实验结果

方法	Lap	DeepWalk	SDNE	TADW	CANE	LANAHS_A	LANAHS
16 维	0.910 9	0.994 8	0.519 6	0.918 7	0.905 4	0.982 5	0.996 9
32 维	0.930 1	0.997 0	0.523 4	0.955 3	0.952 3	0.983 5	0.998 1
48 维	0.952 1	0.997 8	0.524 2	0.966 6	0.957 4	0.983 7	0.999 2
64 维	0.958 6	0.998 2	0.526 7	0.973 0	0.978 7	0.985 4	0.999 2
80 维	0.965 9	0.998 5	0.531 6	0.976 4	0.979 2	0.993 9	0.999 3
96 维	0.975 3	0.998 7	0.533 6	0.978 7	0.985 8	0.994 0	0.999 3
112 维	0.976 8	0.998 8	0.534 1	0.980 5	0.985 9	0.996 5	0.999 3
128 维	0.978 4	0.998 9	0.539 3	0.981 9	0.986 3	0.996 7	0.999 3

表 5　Facebook 数据集链路重构实验结果

方法	Lap	DeepWalk	SDNE	TADW	CANE	LANAHS_A	LANAHS
16 维	0.791 4	0.909 8	0.520 2	0.799 8	0.838 6	0.912 3	0.918 2
32 维	0.811 6	0.927 0	0.570 6	0.828 1	0.857 9	0.917 5	0.940 1
48 维	0.852 9	0.935 1	0.537 6	0.838 0	0.862 4	0.923 4	0.949 7
64 维	0.866 3	0.939 8	0.571 5	0.848 2	0.871 2	0.934 9	0.953 5
80 维	0.876 5	0.942 7	0.611 7	0.855 5	0.884 1	0.942 1	0.955 7
96 维	0.883 8	0.944 7	0.567 3	0.862 4	0.889 7	0.945 6	0.955 4
112 维	0.890 6	0.945 9	0.608 1	0.866 3	0.890 1	0.948 0	0.955 1
128 维	0.895 8	0.946 9	0.561 1	0.871 7	0.894 2	0.947 3	0.9543

5.6　超参数实验

5.6.1　嵌入维度

嵌入向量的维度是表示学习模型中重要的超参数，通常维度增加，可以带来性能提升。固定其他参数后分析各模型在不同维度 d 下的性能，以此衡量模型性能和维度之间的关系。在 Cora 数据集中用链路重构作为实验方案，结果如图 5(a)所示，为使图表更直观，仅展示 DeepWalk、

LANAHS_A 和 LANAHS 3 个最佳效果的模型结果，其他模型性能较差，在同一图中展示会导致趋势不明显。在 DBLP 数据集中，使用节点分类作为实验方案，结果如图 5(b)所示。上述实验结果表明，随着维度的增加模型获得的表示性能也相应增加，整个过程中没有出现较大波动，并且在个维度上都保持了对其他方法的优势，显示出模型的稳定性和良好效果。

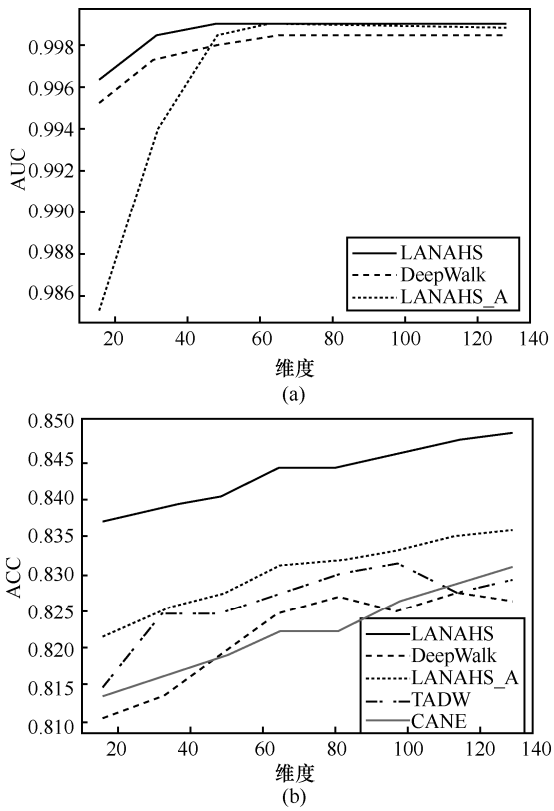


图 5　维度对性能的影响

5.6.2　各项所占比重

模型需要将链路结构、模式结构和属性信息做早期融合后送入神经网络学习，三者信息源在融合前所占的不同比重会对结果有不同影响，比重是指融合前 3 种信息构成向量的维度。选取常见的几种比例进行测试，使用如前文所述的节点分类方法，维度为 128，训练比例为 60%，测试结果见表 6,信息比例对应为链路信息:属性信息:模式结构相似性信息。

表 6　不同信息比例下节点分类的性能

信息比例	Cora	DBLP
1:1:1	83.65%	84.72%
2:2:1	83.05%	85.26%
2:1:1	84.21%	84.21%
2:1:2	83.37%	83.12%

Cora 数据集在 2:1:1 比例下，达到最佳性能，DBLP 数据集在 2:2:1 比例下达到最佳性能，Cora 数据集在 TADW 和 CANE 等属性方法上表现较差，相应地减少文本属性的分配可使模型达到最佳性能，而 DBLP 受文本属性影响较大，这表明信息比例并没有一个明确的答案，对于不同数据集，应该根据实际情况有不同的侧重。

5.6.3　网络深度

通常随着网络深度增加，多层神经网络模型的泛化能力可得到增强，但层数并非越多越好，层数增加可能会带来过拟合问题，同时增加训练难度，使模型难以收敛。本文实验默认采用单隐藏层，可以拟合任何“包含从一个有限空间到另一个有限空间的连续映射”的函数，在真实数据中展现了较好效果。为探究隐藏层层数对模型的影响，本节设置如下实验，选取 Cora 和 DBLP 数据集，固定维度 128 维，隐藏层设置 0 层、1 层、2 层、3 层、4 层 5 种不同情况，训练完成后，选取 50%的节点参与节点分类训练，实验设置和评价指标同第 5.3 节，实验结果见表 7。

表 7　不同隐藏层下节点分类的性能

层数	Cora	DBLP
无隐藏层	82.65%	84.02%
1 层	83.57%	84.89%
2 层	84.24%	85.45%
3 层	84.31%	86.32%
4 层	84.31%	86.15%

通过表 7 可得出结论，随着隐藏层增多，网络获得了更好的表示，这表明增加模型深度可以

提高模型性能。隐藏层为 1 时，模型就可获得比基准方法更好的效果，通过增加网络深度，进一步提高了模型性能，体现采用深度神经网络作为模型架构的有效性。但随隐藏层增加，性能提升效果逐渐不明显，并且训练时间增加，特别地 DBLP 在隐层层数为 4 时出现了性能的小幅下降，因此通常不采用过多的隐藏层。

6　结束语

针对现有网络表示学习方法仅利用显式可见信息学习网络表示的现状，提出了“模式结构”相似性概念，深度挖掘了网络中存在的子图，并对其形状进行表征。利用模式结构相似性，可以挖掘出网络中更多信息，可以更好地表征节点在全局层面的相似性，克服游走所带来的局部局限性。将模式结构相似与属性相似性连接后，与游走所产生的链路结构相似性做融合，用以时时补充低阶相似性带来的不足。在多个真实世界数据集上的实验结果表明，模式结构相似性在节点分类和链路重构方面取得了优异的效果，即所学习到的表示具有更好的性能。

参考文献：

[1] LANDE D, FU M, GUO W, et al. Link prediction of scientific collaboration networks based on information retrieval[J]. World Wide Web, 2020(23): 2239-2257.

[2] FRANCOIS M, DONOVAN P, FONTAINE F. Modulating transcription factor activity: interfering with protein-protein interaction networks[C]//Proceedings of Seminars in Cell & Developmental Biology. Amsterdam: Academic Press, 2020: 12-19.

[3] 李晋, 杨子龙. 微博转发网络中的节点特征和传播模型[J]. 电信科学, 2016, 32(1): 40-45.
LI J, YANG Z L. Node characteristic and propagation model in microblog forwarding network[J]. Telecommunications Science, 2016, 32(1): 40-45.

[4] TSOUMAKAS G, KATAKIS I. Multi-label classification: an overview[J]. International Journal of Data Warehousing and Mining (IJDWM), 2007, 3(3): 1-13.

[5] 周晶, 孙喜民, 于晓昆, 等. 知识图谱与数据应用——智能推荐[J]. 电信科学, 2019, 35(8): 165-172.
ZHOU J, SUN X M, YU X K, et al. Knowledge graph and data application-intelligent recommendation[J]. Telecommunications Science, 2019, 35(8): 165-172.

[6] LIBEN-NOWELL D, KLEINBERG J. The link-prediction problem for social networks[J]. Journal of the American Society for Information Science and Technology, 2007, 58(7): 1019-1031.

[7] 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: ACM Press, 2014: 701-710.

[8] TANG J, QU M, WANG M, et al. Line: Large-scale information network embedding[C]//Proceedings of the 24th International Conference on World Wide Web. [S.l.:s.n.], 2015: 1067-1077.

[9] CAO S, LU W, XU Q. Grarep: learning graph representations with global structural information[C]//Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. New York: ACM Press, 2015: 891-900.

[10] GROVER A, LESKOVEC J. node2vec: scalable feature learning for networks[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2016: 855-864.

[11] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. arXiv: 1301.3781, 2013.

[12] NEWMAN M E J. Modularity and community structure in networks[J]. Proceedings of the National Academy of Sciences, 2006, 103(23): 8577-8582.

[13] WANG D, CUI P, ZHU W. Structural deep network embedding[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2016: 1225-1234.

[14] LI Y, WANG Y, ZHANG T, et al. Learning network embedding with community structural information[C]//Proceedings of IJCAI. San Francisco: Morgan Kaufman, 2019: 2937-2943.

[15] YANG C, LIU Z, ZHAO D, et al. Network representation learning with rich text information[C]//Proceedings of Twenty-Fourth International Joint Conference on Artificial Intelligence. San Francisco: Morgan Kaufman, 2015.

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

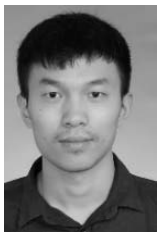
[17] TU C, LIU H, LIU Z, et al. Cane: context-aware network embedding for relation modeling[C]//Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2017: 1722-1731.

[18] HUANG X, SONG Q, LI J, et al. Exploring expert cognition for attributed network embedding[C]//Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining. New York: ACM Press, 2018: 270-278.
[19] HOU C, HE S, TANG K. RoSANE: robust and scalable attributed network embedding for sparse networks[J]. Neurocomputing, 2020(409): 231-243.
[20] JIN D, LI B, JIAO P, et al. Network-specific variational auto-encoder for embedding in attribute networks[C]//Proceedings of IJCAI. San Francisco:Morgan Kaufman, 2019: 2663-2669.
[21] HE Z, LIU J, LI N, et al. Learning network-to-network model for content-rich network embedding[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM Press, 2019: 1037-1045.
[22] LE Q, MIKOLOV T. Distributed representations of sentences and documents[C]//Proceedings of International Conference on Machine Learning. New York: ACM Press, 2014: 1188-1196.
[23] MCCALLUM A K, NIGAM K, RENNIE J, et al. Automating the construction of internet portals with machine learning[J]. Information Retrieval, 2000, 3(2): 127-163.
[24] TANG J, ZHANG J, YAO L, et al. Arnetminer: extraction and mining of academic social networks[C]//Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2008: 990-998.
[25] TRAUD A L, MUCHA P J, PORTER M A. Social structure of facebook networks[J]. Physica A: Statistical Mechanics and its Applications, 2012, 391(16): 4165-4180.
[26] BELKIN M, NIYOGI P. Laplacian eigenmaps and spectral techniques for embedding and clustering[C]//Proceedings of Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2002: 585-591.
[27] PEDREGOSA F, VAROQUAUX G, GRAMFORT A, et al. Scikit-learn: machine learning in Python[J]. Journal of Machine Learning Research, 2011, 12(8): 2825-2830.
[28] SEN P, NAMATA G, BILGIC M, et al. Collective classification in network data[J]. AI Magazine, 2008, 29(3): 93.
[29] HANLEY J A, MCNEIL B J. The meaning and use of the area under a receiver operating characteristic (ROC) curve[J]. Radiology, 1982, 143(1): 29-36.

[作者简介]

邬少清（1995− ），男，宁波大学信息科学与工程学院硕士生，主要研究方向为大数据、数据挖掘。

董一鸿（1969− ），男，博士，宁波大学信息科学与工程学院教授、硕士生导师，主要研究方向为大数据、数据挖掘、人工智能。



王雄（1994− ），男，宁波大学信息科学与工程学院硕士生，主要研究方向为数据挖掘。

曹燕（1993− ），女，宁波大学信息科学与工程学院硕士生，主要研究方向为大数据、数据挖掘。

辛宇（1987− ），男，宁波大学信息科学与工程学院博士生，主要研究方向为数据库、知识工程。