

## Network representation learning: an overview

涂存超, 杨成, 刘知远 and 孙茂松

Citation: 中国科学: 信息科学 47, 980 (2017); doi: 10.1360/N112017-00145

View online: <http://engine.scichina.com/doi/10.1360/N112017-00145>

View Table of Contents: <http://engine.scichina.com/publisher/scp/journal/SSI/47/8>

Published by the [《中国科学》杂志社](#)

---

### Articles you may be interested in

[Ways to sparse representation: An overview](#)

Science in China Series F-Information Sciences 52, 695 (2009);

[An overview of representation theorems for static risk measures](#)

Science in China Series A-Mathematics 52, 1412 (2009);

[An overview of multi-task learning](#)

National Science Review 5, 30 (2018);

[Learning discriminative and invariant representation for fingerprint retrieval](#)

SCIENCE CHINA Information Sciences 62, 019104 (2019);

[Learning block-structured incoherent dictionaries for sparse representation](#)

SCIENCE CHINA Information Sciences 58, 102302 (2015);

---



# 网络表示学习综述

涂存超<sup>1,2,3†</sup>, 杨成<sup>1,2,3†</sup>, 刘知远<sup>1,2,3\*</sup>, 孙茂松<sup>1,2,3</sup>

1. 清华大学计算机科学与技术系, 北京 100084

2. 智能技术与系统国家重点实验室 (清华大学), 北京 100084

3. 清华信息科学与技术国家实验室 (筹), 北京 100084

\* 通信作者. E-mail: liuzy@tsinghua.edu.cn

† 同等贡献

收稿日期: 2017-06-30; 接受日期: 2017-08-01; 网络出版日期: 2017-08-23

国家重点基础研究发展计划 (973) (批准号: 2014CB340501)、国家自然科学基金重大招标项目 (批准号: 13&ZD190)、国家自然科学基金 (批准号: 61572273)、中国科协青年人才托举计划 (批准号: 2016QNRC001) 和清华大学自主科研项目 (批准号: 20151080406) 资助

**摘要** 网络是表达物体和物体间联系的一种重要形式, 针对网络的分析研究的一个关键问题就是研究如何合理地表示网络中的特征信息. 随着机器学习技术的发展, 针对网络中节点的特征学习成为了一项新兴的研究任务. 网络表示学习算法将网络信息转化为低维稠密的实数向量, 并用于已有的机器学习算法的输入. 举例来说, 节点表示可以作为特征送入支持向量机等分类器用于节点分类任务, 也可以作为欧氏空间中的点坐标用于可视化任务. 近年来, 网络表示学习问题吸引了大量的研究者的目光, 本文将针对近年来的网络表示学习工作进行系统性的介绍和总结.

**关键词** 网络, 表示学习, 机器学习, 深度学习, 神经网络

## 1 引言

网络数据形式可以自然地表达物体和物体间的联系, 在我们的日常生活与工作中无处不在. 举例来说, Facebook 和新浪微博等构成了人与人之间的社交网络; 互联网上成千上万个页面构成了网页链接的网络; 国家城市间的运输交通构成了物流网络. 由此可见, 信息网络是我们生产生活中最为常见的一种信息载体和形式. 信息社会中很多网络节点拥有丰富的文本等外部信息, 形成典型的复杂信息网络. 基于复杂信息网络的广泛存在, 对这类网络信息进行研究与分析具有非常高的学术价值和潜在应用价值. 对于复杂信息网络的分析, 根据信息网络载体的不同, 也会具有非常广的普适性.

在学术价值方面, 信息网络是信息的重要表达形式. 随着大数据和深度学习技术的突飞猛进, 人工智能研究正面临新一轮的爆发式发展, 能否对复杂信息网络做出有效合理的数据分析是今后学术研究上的热门话题. 在应用价值方面, 信息网络中拥有着非常广泛的应用场景, 如节点分类、链接预测、

**引用格式:** 涂存超, 杨成, 刘知远, 等. 网络表示学习综述. 中国科学: 信息科学, 2017, 47: 980–996, doi: 10.1360/N112017-00145  
Tu C C, Yang C, Liu Z Y, et al. Network representation learning: an overview (in Chinese). Sci Sin Inform, 2017, 47: 980–996, doi: 10.1360/N112017-00145

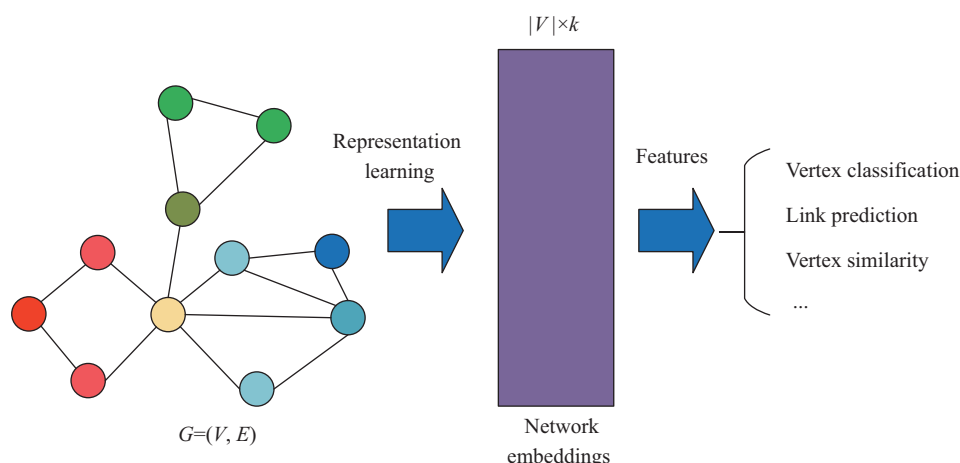


图 1 (网络版彩图) 网络表示学习流程图

Figure 1 (Color online) The flow chart of NRL

社区发现、推荐系统等任务, 具有实际的应用价值. 综上所述, 针对复杂信息网络的研究与应用是人工智能学科发展的学术前沿问题, 是智能信息处理和服务发展的基础技术保障, 针对大规模复杂信息网络的学习研究十分必要.

在类似 Facebook 和新浪微博等社交网络的快速发展下, 很多研究者着力于在网络数据上设计快速有效的算法. 在有关网络的研究中, 一个重要的问题就是如何合适的表示网络信息. 传统的网络表示一般使用高维的稀疏向量. 但是高维稀疏的表示也成为了人们使用统计学习方法时的局限所在, 因为高维的向量将会花费更多的运行时间和计算空间. 随着表示学习技术在自然语言处理等领域的发展和广泛应用, 研究者们转而探索将网络中的节点表示为低维稠密的向量表示的方法. 直觉上来看, 在网络中拓扑结构相似的节点也应该具有相近的向量表示. 这里向量表示的相似性一般用向量间的余弦距离或者欧氏距离来表示. 之后这些作为节点表示的向量就可以用作对应节点的特征并应用到后续的任务场景中.

基于网络的表示学习研究旨在探索能够更好地研究分析复杂信息网络中的节点间的联系, 寻找解决信息网络背景下的各种实际问题的普适方法, 有效融合网络结构与节点外部信息, 形成更具区分性的网络表示. 近年来, 网络表示学习问题吸引了大量的研究者的目光, 相关的论文工作也层出不穷, 本文将针对近年来的网络表示学习工作进行系统性的介绍和总结.

## 2 网络表示学习的定义

本节将形式化地介绍网络表示学习的意义. 如图 1 所示, 网络表示是衔接网络原始数据和网络应用任务的桥梁. 网络表示学习算法负责从网络数据中学习得到网络中每个节点的向量表示, 之后这些节点表示就可以作为节点的特征应用于后续的网络应用任务, 如节点分类、链接预测和可视化等.

将网络记作  $G = (V, E)$ , 其中  $V$  是节点集合,  $E$  是边的集合. 边  $e = (v_i, v_j) \in E$  表示了节点  $v_i$  到  $v_j$  的一条边. 网络的邻接矩阵定义为  $A \in \mathbb{R}^{|V| \times |V|}$ , 其中  $A_{ij} = 1$  如果  $(v_i, v_j) \in E$ , 否则  $A_{ij} = 0$ . 邻接矩阵是网络数据的一种简单直接的表达形式. 邻接矩阵  $A$  的每一行表示了一个节点和所有其他节点的链接关系, 可以看作是对应节点的一种表示.

虽然方便直接, 使用邻接矩阵的网络表示受到计算效率问题的影响. 邻接矩阵  $A$  占用了  $|V| \times |V|$  的存储空间, 这在  $|V|$  增长到百万级时通常是不可接受的. 另一方面, 邻接矩阵中绝大多数是 0, 数据十分稀疏. 这种数据稀疏性使得快速有效的统计学习方法的应用变得困难<sup>[1]</sup>.

因此, 研究者们转而为网络中的节点学习低维稠密的向量表示. 形式化地, 网络表示学习的目标就是对每个节点  $v \in V$  学习一个实数向量  $R_v \in \mathbb{R}^k$ , 其中向量的维度  $k$  远远小于节点的总个数  $|V|$ . 网络表示学习的过程可以是无监督或者半监督的. 通过优化算法自动得到而不需要特征工程的节点表示可以进一步用于后续的网络应用任务, 如节点分类. 这些低维的向量表示使得快速高效的算法设计成为可能, 而不必再去考虑原本的网络结构.

### 3 基于网络结构的网络表示学习

随着机器学习技术的发展, 对于网络中节点的特征学习已经成为了一项非常重要的任务. 网络表示学习算法将网络信息转化为低维稠密的实数向量, 并将其用作已有的机器学习算法的输入. 举例来说, 节点的表示可以作为特征, 送到类似支持向量机的分类器中. 同时, 节点表示也可以转化成空间坐标, 用于可视化任务<sup>[2~5]</sup>. 下面将介绍已有的网络表示学习算法和相关进展.

#### 3.1 基于矩阵特征向量计算

较早的用于网络表示学习的算法主要归于此类. 谱聚类算法通过计算关系矩阵的前  $k$  个特征向量或奇异向量来得到  $k$  维的节点表示. 关系矩阵一般就是网络的邻接矩阵或者 Laplace 矩阵. 这类方法强烈的依赖于关系矩阵的构建, 不同的关系矩阵的评测结果差异很大. 一般来讲, 基于谱聚类方法的时间复杂度较高, 因为特征向量和奇异向量的计算时间是非线性的. 另一方面, 谱聚类方法需要将关系矩阵整体存于内存之中, 所以空间复杂度也是不能忽略的. 这些局限性阻止了这类算法在大规模数据和在线平台上的扩展应用. 现在将展示几种谱聚类算法的实例. 这些方法的适用性如表 1 所示.

局部线性表示 (locally linear embedding)<sup>[6,7]</sup> 假设节点的表示是从同一个流形中采样得到的. 局部线性表示假设一个节点和它邻居的表示都位于该流形的一个局部线性的区域. 也就是说, 一个节点的表示可以通过它的邻居节点的表示的线性组合来近似得到. 局部线性表示使用邻居节点表示的加权和与中心节点表示的距离作为损失函数. 最小化损失函数的优化问题最终转化成某个关系矩阵特征向量计算问题求解.

Laplace 特征表 (Laplace eigenmap)<sup>[8,9]</sup> 简单的假设两个相连的节点的表示应该相近. 特别地, 这里表示相近是由向量表示的欧氏距离的平方来定义. 该优化问题可以类似地转化为 Laplace 矩阵的特征向量计算问题.

有向图表示 (directed graph embedding)<sup>[10]</sup> 进一步扩展了 Laplace 特征表方法, 给不同点的损失函数以不同的权重. 其中点的权重是由基于随机游走的排序方法来决定, 如 PageRank.

不同于前面的方法, Tang 和 Liu<sup>[11]</sup> 假设节点表示的每一维对应着该节点从属于一个社区 (community) 的强度, 进而将模块性 (modularity) 引入了损失函数. 模块性是衡量网络分离程度的指标, 高的模块性值意味着在同一模块内节点连接紧密, 而不同模块间节点连接稀疏. 最终该优化问题转化为了归一化的 Laplace 矩阵的特征向量计算. 此类方法一般先定义一个关于节点表示的线性或二次损失函数. 然后将最优化问题转化为某个关系矩阵的特征向量计算问题. 这一类方法最主要的缺点在于复杂度: 大规模矩阵的特征向量计算是非常消耗计算时间和空间的.

表 1 局部线性表示 (LLE)、Laplace 特征表 (Laplace eigenmap)、有向图表示 (DGE) 的适用性比较

Table 1 The comparison of LLE, Laplace eigenmap, and DGE

Model	Graph type		
	Undirected	Weighted	Directed
LLE	✓	—	—
Laplace eigenmap	✓	✓	—
DGE	✓	✓	✓

表 2 DeepWalk 算法和 word2vec 的类比

Table 2 The comparison between DeepWalk and word2vec

Model	Target	Input	Output
Word2vec	Words	Sentences	Word embeddings
DeepWalk	Nodes	Node sequences	Node embeddings

### 3.2 基于简单神经网络的算法

3.1 小节介绍的方法中, 对最优化问题求最优解的过程, 如特征向量的计算, 对于大规模的网络数据来说是非常耗时的. 另一方面, 基于神经网络的方法已经在自然语言和图像处理领域上取得了非常突出的成果. 虽然梯度下降的参数更新方法无法保证总是得到最优化问题的最优解, 但是神经网络的方法一般更加快速而且也能得到相当不错的结果.

DeepWalk 算法<sup>[1]</sup>第一次将深度学习中的技术引入到网络表示学习领域. DeepWalk 算法充分利用了网络结构中的随机游走序列的信息. 使用随机游走序列而不是邻接矩阵的优势有两点: 首先, 随机游走序列只依赖于局部信息, 所以可以适用于分布式和在线系统, 而使用邻接矩阵就必须把所有信息存储于内存中处理, 面临着较高的计算时间和空间消耗. 第二, 对随机游走序列进行建模可以降低建模 0-1 二值邻接矩阵的方差和不确定性.

无监督表示学习方法在自然语言处理领域已经得到了广泛的学习与应用. Perozzi 等<sup>[1]</sup>通过实验验证了随机游走序列中节点和文档中的单词一样都遵从指数定律 (power-law), 从而进一步将著名的词表示学习算法 word2vec<sup>[12~14]</sup>应用在随机游走序列上, 学习节点表示.

形式化地, 假设生成的随机游走序列中  $v_i$  的左右为  $w$  的窗口中的序列是  $v_{i-w}, \dots, v_{i-1}, v_{i+1}, \dots, v_{i+w}$ . DeepWalk 算法会计算由中心节点  $v_i$ , 产生两侧节点的概率. 作为模型简化, Skip-Gram 模型忽略了节点的顺序和距离中心节点的距离. 损失函数可以形式化地由式 (1) 表示:

$$\min_R \sum_{k=-w, k \neq 0}^w -\log \Pr(v_{i+k}|R_i), \quad (1)$$

其中  $\Pr(v_{i+k}|R_i)$  由式 (2) 定义的 softmax 函数得到:

$$\Pr(v_j|R_i) = \frac{\exp(R_i C_j^T)}{\sum_{k=1}^{|V|} \exp(R_i C_k^T)}. \quad (2)$$

表 2 总结了 DeepWalk 算法和 word2vec 的类比情况. DeepWalk 首先在网络上采样生成大量的随机游走序列, 然后用 Skip-gram 和 Hierarchical Softmax 模型对随机游走序列中每个局部窗口内的节点对进行概率建模, 最大化随机游走序列的似然概率, 并最终使用随机梯度下降学习参数.

表 3 不同网络表示学习方法的比较  
Table 3 The comparison of various NRL methods

Method	Matrix	Accuracy	Speed	Performance
Spectral clustering	L	Precise	Fast	Low
DeepWalk	$\sum_{k=1}^K \frac{A^k}{K}$	Approximate	Fast	Medium
GraRep	$A^k, k = 1, \dots, K$	Precise	Slow	High

基于简单神经网络的另一个代表性的网络表示学习算法就是 LINE 算法<sup>[15]</sup>. Tang 等<sup>[15]</sup>提出了一种可以适用于大规模的有向带权图的网络表示学习算法. 为了对节点间的关系进行建模, LINE 算法用观察到的节点间连接刻画了第一级相似度关系, 用不直接相连的两个节点的共同邻居刻画了这两个点之间的第二级相似度关系. 直觉上说, 对直接相连的两个节点间关系的刻画等价于对原始网络的邻接矩阵的建模. 但是一个网络中的边关系往往是非常稀疏的, 所以有必要进一步刻画第二级相似度关系来考虑虽然并不直接相连, 但是共同邻居较多的节点对, 从而对第一级相似度的信息予以补充.

具体来说, LINE 算法对所有的第一级相似度和第二级相似度节点对进行了概率建模, 并最小化该概率分布和经验分布之间的 KL 距离. 参数学习由随机梯度下降算法决定.

### 3.3 基于矩阵分解的方法

给定关系矩阵, 对关系矩阵进行矩阵分解达到降维的效果, 从而得到节点的网络表示. Yang 等<sup>[16]</sup>证明了 DeepWalk 算法实际上等价于某个特定关系矩阵的矩阵分解. 由此可见, 矩阵分解算法也是学习网络表示的一种重要手段.

GraRep 算法<sup>[17]</sup>考虑了一种特别的关系矩阵. GraRep 通过 SVD 分解对该关系矩阵进行降维从而得到  $k$  步网络表示. 形式化地, 假设首先对邻接矩阵  $A$  进行行归一化处理, 使得矩阵  $A$  中每行的和等于 1. 则 GraRep 算法在计算  $k$  步网络表示时分解了矩阵  $A^k$ , 该关系矩阵中的每个单元对应着两个节点间通过  $k$  步的随机游走抵达的概率. 更进一步, GraRep 尝试了不同的  $k$  值, 并将不同  $k$  值对应的  $k$  步网络表示拼接起来, 组成维度更高、表达能力也更强的节点表示. 但 GraRep 的主要缺点在于其在计算关系矩阵  $A^k$  的时候计算效率很低.

Yang 等<sup>[18]</sup>在其后续工作中将基于矩阵分解或者可以转化为矩阵分解的方法总结成同一个算法框架: 第一步构建节点间的关系矩阵, 第二步对该矩阵进行矩阵分解操作得到网络表示. 该工作将谱聚类方法<sup>[11]</sup>, DeepWalk 和 GraRep 方法第一步构建关系矩阵的过程进行了分析对比, 并总结在表 3 中, 其中  $L$  是 Laplace 矩阵.

通过对表 3 的观察, 可以得出两个结论. 一是对更高阶的关系矩阵  $A^k$  的构建可以提升网络表示的效果, 二是精确的计算高阶的关系矩阵  $A^k$  计算效率很低. 这两个结论促使我们寻找一种可以间接近似高阶的关系矩阵且不增加计算复杂度的方法.

Yang 等<sup>[18]</sup>提出了一种简单的网络表示更新策略 NEU, 如式 (3) 所示:

$$R_{\text{new}} = R + \lambda_1 A \cdot R + \lambda_2 A \cdot (A \cdot R), \quad (3)$$

其中  $\lambda_1, \lambda_2$  是超参数, 一般设置为 0.5 和 0.25. 该工作证明了式 (3) 中的策略可以让更新后的网络表示近似等价于从更高阶的关系矩阵中分解而来, 而不增加计算复杂度. 实际上, 当该算法作用于 DeepWalk 算法的输出结果时, 只占用 DeepWalk 算法 1% 的时间, 就可以有非常显著的提升效果.

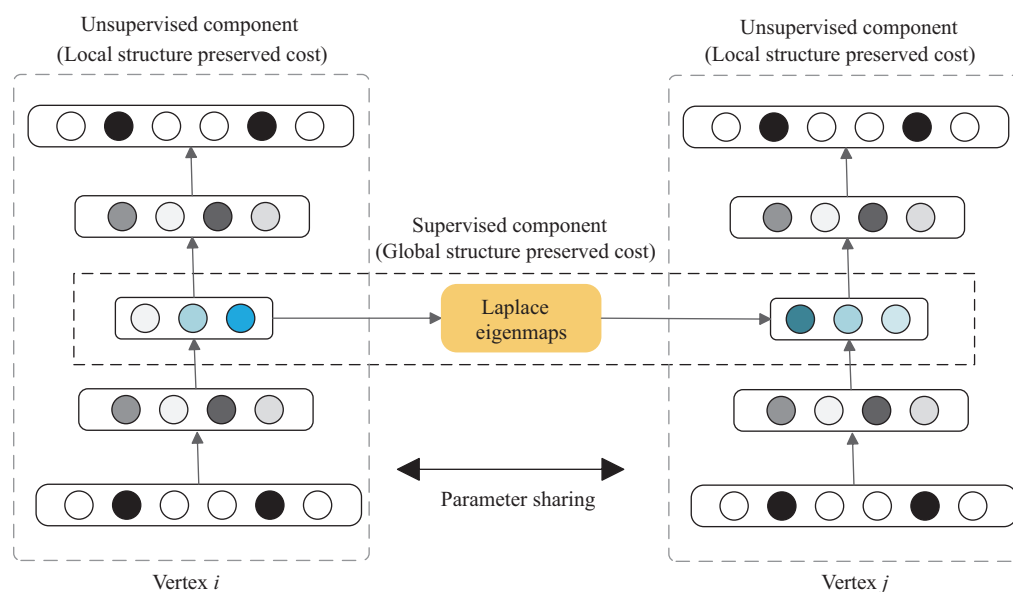


图 2 (网络版彩图) SDNE 算法 (修改自文献 [19])

Figure 2 (Color online) The framework of SDNE (modified from [19])

### 3.4 基于深层神经网络的方法

和之前使用浅层神经网络的方法不同, SDNE<sup>[19]</sup> 使用深层神经网络对节点表示间的非线性进行建模. 整个模型可以被分为两个部分: 一个是由 Laplace 矩阵监督的建模第一级相似度的模块, 另一个是由无监督的深层自编码器对第二级相似度关系进行建模. 最终 SDNE 算法将深层自编码器的中间层作为节点的网络表示, 模型如图 2 所示.

### 3.5 基于社区发现的算法

如谱聚类算法中展示的, 研究者已经考虑从社区发现角度学习网络表示. 具体来说, 就是让节点表示的每一维对应该节点从属于一个社区的强度, 然后设计最优化目标进行求解. 这类算法会学习得到上述的社区强度关系表示, 然后转化为社区发现的结果. 而学习社区强度关系表示的过程可以看作是无监督的非负节点表示学习的过程.

BIGCLAM<sup>[20]</sup> 作为一个可覆盖社区发现算法, 为每个网络中的节点学习了一个上述的  $k$  维非负向量表示. BIGCLAM 算法对网络中每条边的生成概率进行建模: 两个点的向量表示内积越大, 那么这两个点之间形成边的概率也就越高. 算法的最大化目标是整个网络结构的最大似然概率. 最优化求解参数的过程由随机梯度下降算法实现.

### 3.6 保存特殊性质的网络表示

使用向量表示代替原始网络的策略在带来便利的同时, 也会丢失很多原始网络中的信息. 比如大多数网络表示学习方法使用向量表示间的内积或者余弦距离刻画节点相似度. 但内积或者余弦距离都是无向的, 会丢失网络中的非对称性. 另一方面, 一些依赖于网络结构定义的性质, 如社区 (community) 等信息, 也会在网络表示学习的过程中丢失.



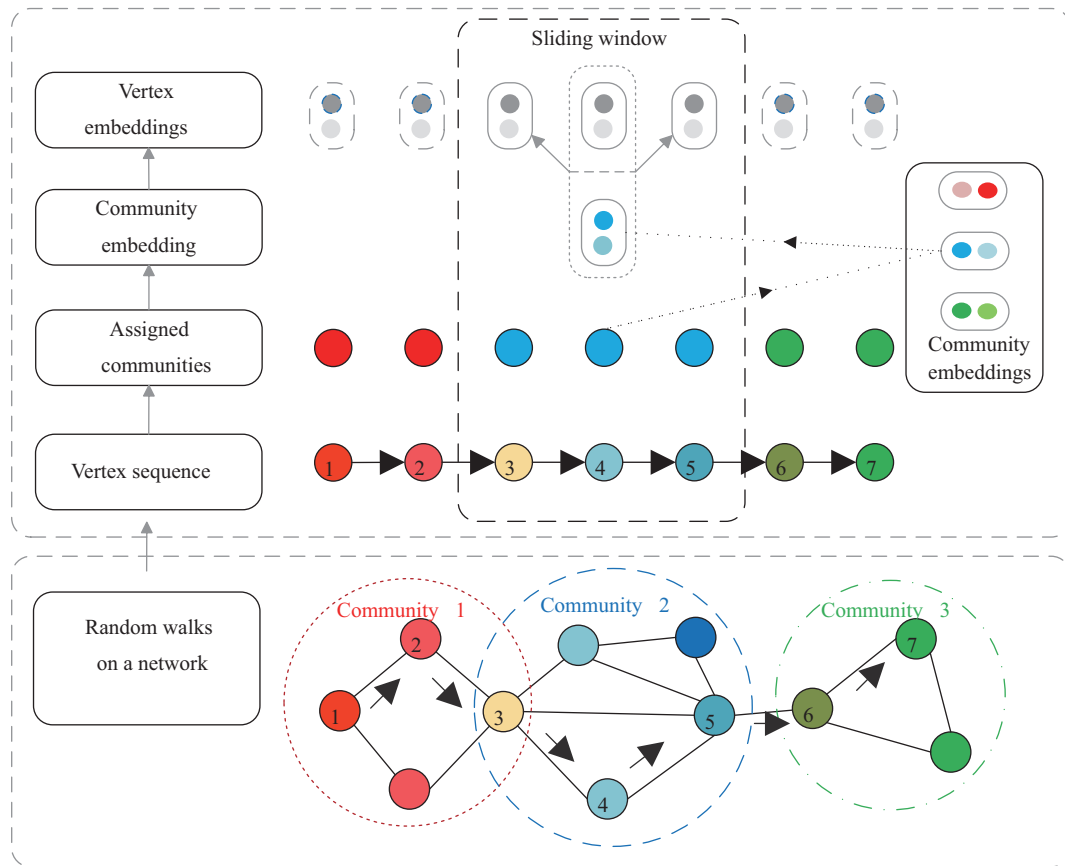


图 3 (网络版彩图) CNRL 算法 (修改自文献 [22])

Figure 3 (Color online) The framework of CNRL (modified from [22])

HOPE 算法<sup>[21]</sup>为每个节点刻画了两种不同的表示,并着眼于保存原始网络中的非对称性信息. HOPE 构建了不同的非对称的关系矩阵,然后使用 JDGSVD 算法进行矩阵降维得到节点的网络表示.

CNRL 算法<sup>[22]</sup>考虑了在节点表示中嵌入网络隐藏的社区信息.如图 3 所示, CNRL 假设每个节点属于多个社区,也就是每个节点在所有的社区上有一个概率分布. DeepWalk 将随机游走生成的节点序列看作句子,将序列中的节点看作文本中的词,直接用训练词向量的 Skip-Gram 模型来训练节点向量.受这种类比的启发, CNRL 将网络中的社区看作文本中的主题,也就是说,网络中相关的节点倾向于行程社区,而文本中相关的词则会构成主题.因此, CNRL 算法在生成的随机游走序列上,将每个节点序列看成一篇文档,通过基于 Gibbs 采样的 LDA<sup>[23]</sup>来学习每个节点的社区分布,并通过随机采样的方式,来给序列中的节点分配其对应的社区标签.随后,在 Skip-Gram 模型的基础上,用中心节点的节点表示和对应的社区表示同时去预测随机游走序列中的邻近节点,从而将社区结构信息保存在节点表示中.

为了对通过主题模型检测出的社区有一个直观的感受,验证将网络中的社区类比为文本中的主题的正确性, CNRL 对一个小的 Karate 网络进行了社区发现结果的可视化,如图 4 所示.可以发现, CNRL 能够有效检测出不同规模的有重叠的社区,以及有效的识别出社区边界.



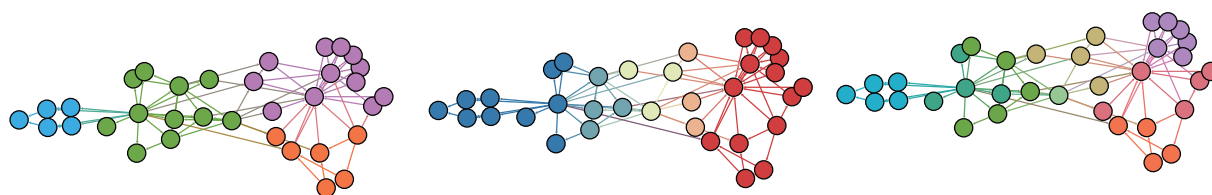


图 4 (网络版彩图) Karate 网络社区发现结果可视化 (快速展开算法、CNRL-2、CNRL-4) (修改自文献 [22])

Figure 4 (Color online) Community detection results on Karate (Fast unfolding, CNRL-2, CNRL-4) (modified from [22])

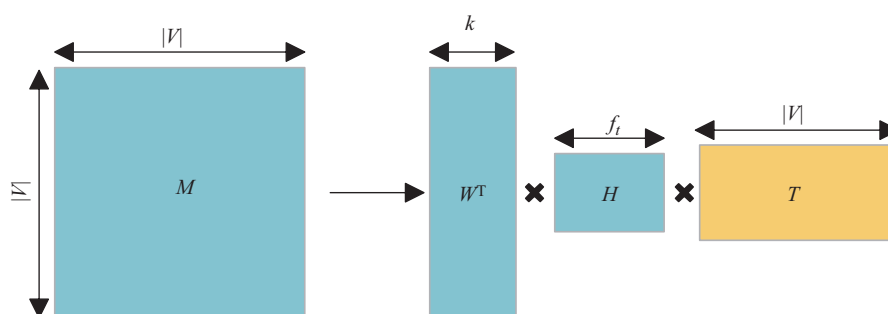


图 5 (网络版彩图) TADW 算法 (修改自文献 [16])

Figure 5 (Color online) The framework of TADW (modified from [16])

## 4 结合外部信息的网络表示学习

真实世界中的网络节点往往会伴随着丰富的外部信息, 例如文本信息、标签类别信息等. 而传统网络表示学习主要依赖于网络拓扑结构信息, 而忽略了这些异质的外部信息. 因此, 如何能够在网络表示学习过程中, 考虑这些外部信息, 提高网络表示的质量和增强表示向量在网络分析任务上的效果, 是网络表示学习领域的重要挑战.

### 4.1 结合文本信息的方法

在网络数据中, 除去节点间的边信息以外, 也会有很多依存于网络的文本信息. 比如社交网络中, 除去用户间的好友关系, 也会有丰富的用户状态或者博客内容等文本信息. 我们可以利用这些文本信息作为网络结构信息的补充, 进一步增强网络节点表示的强度和效果.

TADW<sup>[16]</sup> 在矩阵分解框架下, 将节点的文本特征引入网络表示学习. 如图 5 所示, TADW 算法基于矩阵分解形式的 DeepWalk 算法进一步加强得到: 将关系矩阵  $M$  分解成 3 个小的矩阵乘积, 其中黄色矩阵  $T$  是固定的文本特征向量, 另外两个矩阵是参数矩阵. TADW 算法使用共轭梯度下降法迭代更新  $W$  矩阵和  $H$  矩阵求解参数.

真实世界中的网络节点在与其他节点进行交互时, 往往会展现出不同方面的特点. 例如, 一个研究者与不同的研究者发生合作关系往往因为不同的研究主题; 社交媒体中的用户会因为不同的原因与其他用户建立联系. 然而, 已有的网络表示学习方法会给每个网络节点学习一个固定的表示向量, 不能展现出同一节点对于不同邻居节点角色的变化. 此外, 这些方法不能对节点之间的关系进行有效的建模和解释. 因此, CANE<sup>[24]</sup> 利用网络节点的文本信息来对节点之间的关系进行解释, 来为网络节点根据不同的邻居学习上下文相关的网络表示.

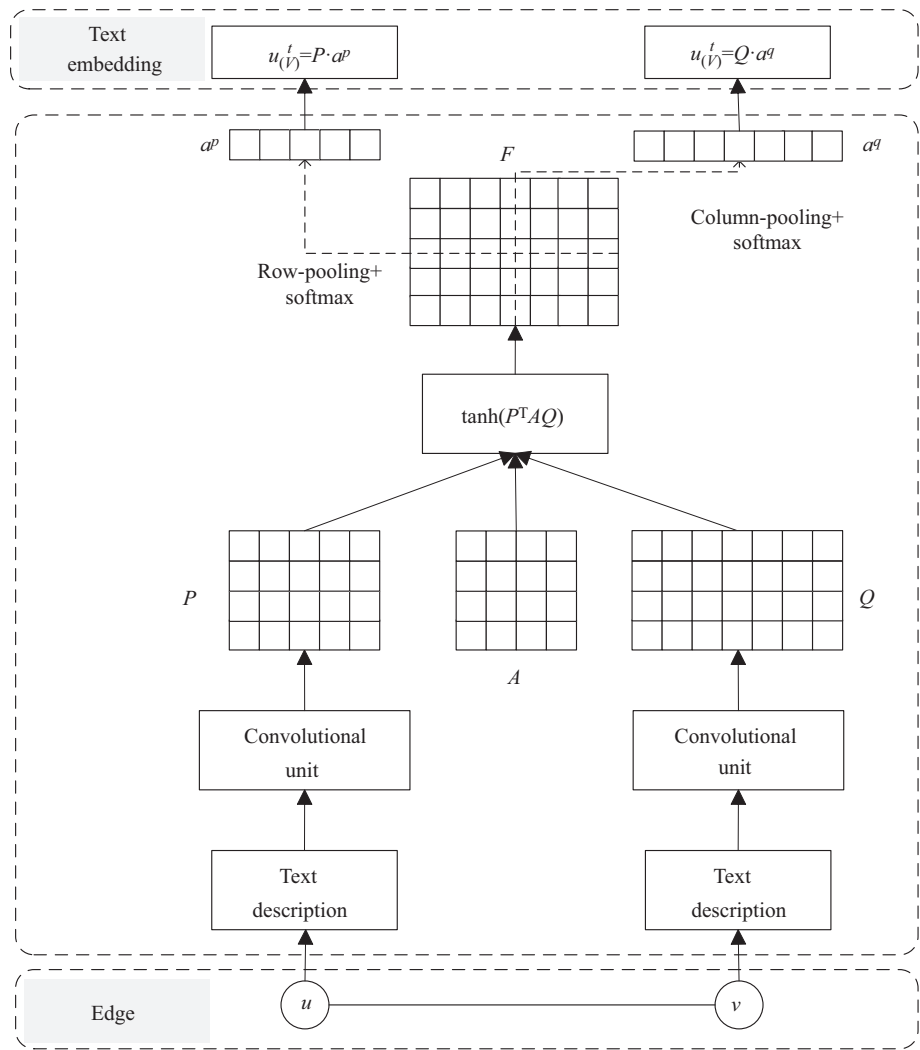


图 6 CANE 算法 (修改自文献 [24])

Figure 6 The framework of CANE (modified from [24])

CANE 假设每个节点的表示向量由文本表示向量及结构表示向量构成, 其中, 文本表示向量的生成过程与边上的邻居相关, 所以生成的节点表示也是上下文相关的. 如图 6 所示, CANE 利用卷积神经网络对一条边上两个节点的文本信息进行编码. 在文本表示生成的过程中, 利用相互注意力机制, 选取两个节点彼此最相关的卷积结果构成最后的文本表示向量.

4.2 半监督的网络表示学习

无监督的网络表示学习中, 其后续任务很多是以节点表示作为特征的节点分类任务. 之前的工作主要基于无监督的网络表示学习, 在针对节点分类等机器学习任务时, 缺少区分性. 半监督的网络表示学习的想法就是把已经标注的节点的节点类别或者标签利用起来, 加入到网络表示学习的过程中, 从而针对性的提升节点网络表示在后续分类任务中的效果.

为了解决这个问题, Tu 等 [25] 提出了一种半监督的网络表示学习方法 MMDW, 来学习有区分性

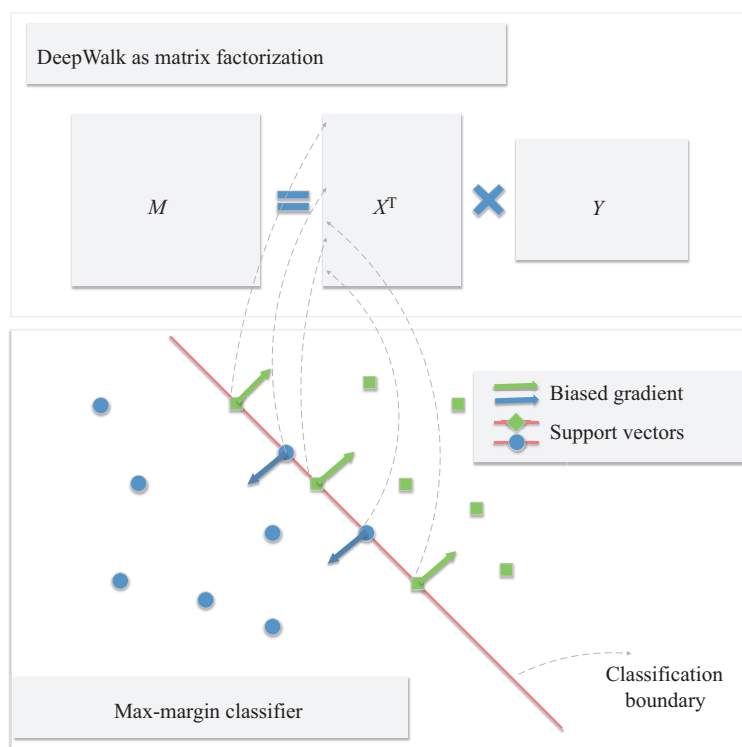


图 7 (网络版彩图) MMDW 算法 (修改自文献 [25])

Figure 7 (Color online) The framework of MMDW (modified from [25])

的网络表示. 如图 7 所示, MMDW 同时学习矩阵分解形式的网络表示模型和最大间隔分类器. 为了增大网络表示的区分性, MMDW 会针对分类边界上的支持向量计算其偏置向量, 使其在学习过程中向正确的类别方向进行偏置, 从而增大表示向量的区分能力.

受最大间距的分类器影响, 该模型学习得到的节点向量不仅包含网络结构的特征, 也会拥有分类标签带来的区分性. 在节点分类任务上的实验效果证明了 MMDW 模型使用半监督训练表示的优势. 此外, 在网络表示的可视化中 (如图 8), 也可观察到节点之间的区分性较无监督模型更加明显. DDRW<sup>[26]</sup> 也采用了类似的方式, 同时训练 DeepWalk 模型和最大间隔分类器, 来提高网络节点分类的效果.

Node2vec<sup>[27]</sup> 通过改变随机游走序列生成的方式进一步扩展了 DeepWalk 算法. DeepWalk 选取随机游走序列中下一个节点的方式是均匀随机分布的. 而 node2vec 通过引入两个参数  $p$  和  $q$ , 将宽度优先搜索和深度优先搜索引入了随机游走序列的生成过程. 宽度优先搜索注重邻近的节点并刻画了相对局部的一种网络表示, 宽度优先中的节点一般会出现很多次, 从而降低刻画中心节点的邻居节点的方差; 深度优先搜索反应了更高层面上的节点间的同质性.

特别地, node2vec 中的两个参数  $p$  和  $q$  控制随机游走序列的跳转概率, 如图 9 所示, 假设上一步游走的边为  $(t, v)$ , 那么对于节点  $v$  的不同邻居, node2vec 根据  $p$  和  $q$  定义了不同的邻居的跳转概率,  $p$  控制跳向上一个节点的邻居的概率,  $q$  控制跳向上一个节点的非邻居的概率, 具体的未归一的跳转概

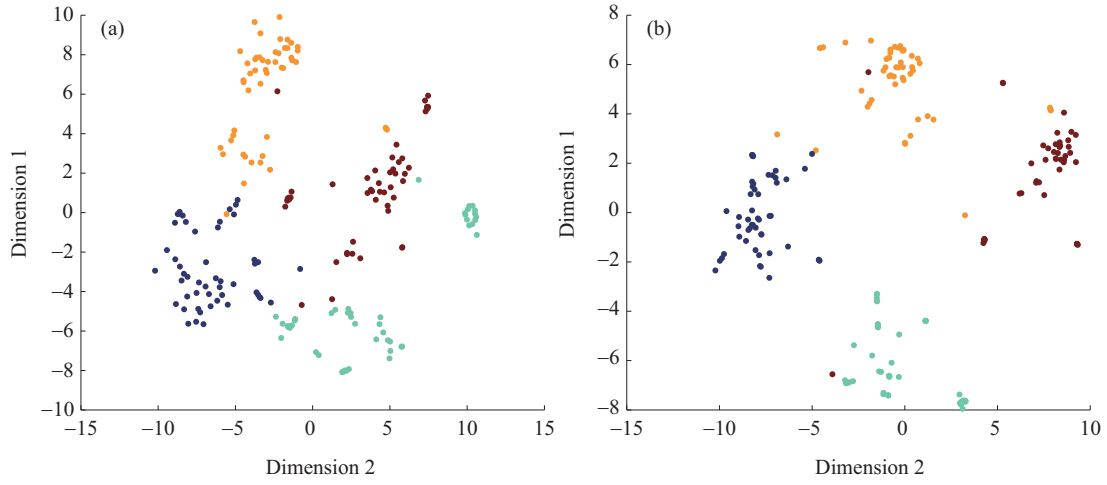


图 8 (网络版彩图) DeepWalk 和 MMDW 可视化结果 (修改自文献 [14])

Figure 8 (Color online) The visualization results of (a) DeepWalk and (b) MMDW (modified from [14])

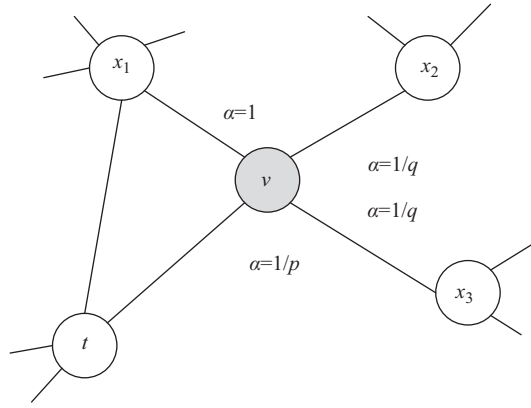


图 9 Node2vec 算法 (修改自文献 [27])

Figure 9 The framework of node2vec (modified from [27])

率值  $\pi_{vx} = \alpha_{pq}(t, x)$  如下所示:

$$\alpha_{pq}(t, x) = \begin{cases} \frac{1}{p}, & \text{if } d_{tx} = 0, \\ 1, & \text{if } d_{tx} = 1, \\ \frac{1}{q}, & \text{if } d_{tx} = 2. \end{cases} \quad (4)$$

其中,  $d_{tx}$  表示节点  $t$  和  $x$  之间的最短距离. 为了获得最优的超参数  $p$  和  $q$  的取值, node2vec 通过半监督形式, 利用网格搜索最合适的参数学习节点表示.

其他的半监督网络表示学习方法还包括: GCN [28] 设计了一种作用于网络结构上的卷积神经网络, 并使用一种基于边的标签传播规则实现半监督的网络表示学习. Planetoid [29] 联合地预测一个节点的邻居节点和类别标签, 类别标签同时取决于节点表示和已知节点标签, 从而进行半监督表示学习.

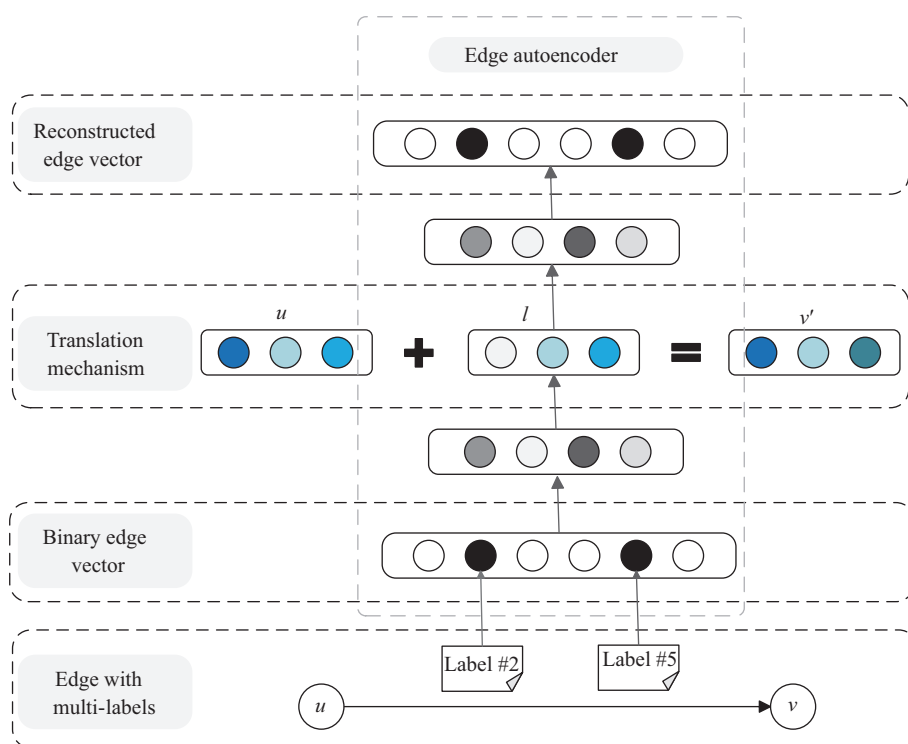


图 10 (网络版彩图) TransNet 算法 (修改自文献 [30])

Figure 10 (Color online) The framework of TransNet (modified from [30])

### 4.3 结合边上标签信息的网络表示学习

除了节点本身附加的文本、标签等信息外,节点与节点之间也存在着丰富的交互信息.例如,社交媒体上用户之间会存在交谈、转发等文本信息;论文合作网络中,研究者之间存在合作的论文的具体信息.然而,已有的网络表示学习模型更侧重于节点本身的信息,而把边简单地看作 0, 1 值或者连续的实值,而忽略边上丰富的语义信息.同时,已有的网络表示学习一般采用节点分类、链接预测等网络分析任务来衡量网络表示学习的质量,而忽略了对节点之间具体关系的建模和预测能力.

为了解决关系的建模和预测问题, Tu 等<sup>[30]</sup>提出了 TransNet 模型,利用平移机制来解决社会关系抽取问题.如图 10 所示, TransNet 假设头结点表示向量加上关系表示向量等于尾结点表示向量.其中,通过关键词抽取、命名实体识别等方式,对交互文本抽取出标签集合来表示关系.随后,通过深层自动编码器对标签集合进行压缩,来得到关系的表示向量.该模型能够有效地预测未标注的边上的标签集合,在社会关系抽取任务上取得了显著的提升.

## 5 评测任务和应用场景

### 5.1 节点分类

在进行网络数据的分析时,一个最常见的场景就是对网络中的节点进行合理的划分.举例来说,在社交网络上,不同的用户可以根据他们的兴趣爱好不同分为不同的类别.然而实际数据中的类别标注信息是十分稀疏的,所以需要设计算法利用节点间的连接关系以及少量的已标注分类信息,对大量

表 4 Cora 数据集上的分类结果

Table 4 Classification results on Cora

	Accuracy (%)			Time (s)
	10% <sup>a)</sup>	50% <sup>a)</sup>	90% <sup>a)</sup>	
GF	50.8 (68.0)	61.8 (77.0)	64.8 (77.2)	4 (+0.1)
SC	55.9 (68.7)	70.8 (79.2)	72.7 (80.0)	1 (+0.1)
DeepWalk	71.3 (76.2)	76.9 (81.6)	78.7 (81.9)	31 (+0.1)
LINE <sub>1st</sub>	64.8 (70.1)	76.1 (80.9)	78.9 (82.2)	62 (+0.1)
LINE <sub>2nd</sub>	63.3 (73.3)	73.4 (80.1)	75.6 (80.3)	67 (+0.1)
node2vec	76.9 (77.5)	81.0 (81.6)	81.4 (81.9)	56 (+0.1)
TADW	78.1 (84.4)	83.1 (86.6)	82.4 (87.7)	2 (+0.1)
GraRep	70.8 (76.9)	78.9 (82.8)	81.8 (84.0)	67 (+0.3)

a) The number means training ratio.

表 5 BlogCatalog 数据集上的分类结果

Table 5 Classification results on BlogCatalog

	Macro-F1 (%)			Micro-F1 (%)			Time (s)
	1% <sup>a)</sup>	5% <sup>a)</sup>	9% <sup>a)</sup>	1% <sup>a)</sup>	5% <sup>a)</sup>	9% <sup>a)</sup>	
GF	6.6 (7.9)	9.8 (11.3)	10.3 (12.2)	17.0 (19.6)	22.2 (25.0)	23.7 (26.7)	19 (+1)
SC	8.4 (9.3)	13.1 (14.8)	14.5 (17.0)	19.4 (20.3)	26.9 (28.1)	29.0 (31.0)	10 (+1)
DeepWalk	12.4 (13.6)	18.3 (20.1)	20.4 (22.0)	24.9 (26.4)	31.5 (33.7)	33.7 (35.9)	935 (+1)
LINE <sub>1st</sub>	11.1 (12.2)	16.6 (18.3)	18.6 (20.1)	23.1 (24.7)	29.3 (31.6)	31.8 (33.5)	241 (+1)
LINE <sub>2nd</sub>	10.3 (11.2)	15.0 (16.8)	16.5 (18.3)	21.5 (25.0)	27.9 (31.6)	30.0 (33.6)	244 (+1)
node2vec	12.5 (13.0)	19.2 (19.8)	21.9 (22.5)	25.0 (27.0)	31.9 (34.5)	35.1 (37.2)	454 (+1)

a) The same as in Table 4.

表 6 Flickr 数据集上的分类结果

Table 6 Classification results on Flickr

	Macro-F1 (%)			Micro-F1 (%)			Time (s)
	1% <sup>a)</sup>	5% <sup>a)</sup>	9% <sup>a)</sup>	1% <sup>a)</sup>	5% <sup>a)</sup>	9% <sup>a)</sup>	
GF	4.3 (5.2)	4.9 (5.4)	5.0 (5.4)	21.1 (21.8)	22.0 (23.1)	21.7 (23.4)	241 (+8)
SC	8.6 (10.9)	11.6 (14.3)	12.3 (15.0)	24.1 (29.2)	27.5 (34.1)	28.3 (34.7)	102 (+8)
DeepWalk	10.5 (11.6)	17.1 (17.8)	19.1 (19.8)	31.8 (33.1)	36.3 (36.7)	37.3 (37.6)	9,292 (+8)
LINE <sub>1st</sub>	10.3 (10.7)	16.0 (16.6)	17.6 (18.2)	32.0 (32.7)	35.9 (36.4)	36.8 (37.2)	2,664 (+8)
LINE <sub>2nd</sub>	7.8 (8.5)	13.1 (13.5)	14.7 (15.2)	30.0 (31.0)	34.2 (34.4)	35.1 (35.2)	2,740 (+8)

a) The same as in Table 4.

的未标注节点进行分类情况进行标注. 类似的任务场景还有对互联网上的各个网页进行内容上的分类. 在节点分类的应用上, 根据算法预测的未标注节点的类别可以为节点的标签进行推荐, 节约手工标注的人力成本, 在各类网络场景都有非常实际的应用. 表 4 ~ 6 列出了不同的网络表示学习方法在 Cora, BlogCatalog 和 Flickr 3 个公开数据集上的节点分类效果对比. 括号中的数字代表着 NEU 更新后的分类效果. 时间一列记录了各算法的运行时间, 括号中的数字代表了 NEU 更新算法所花费的额外运行

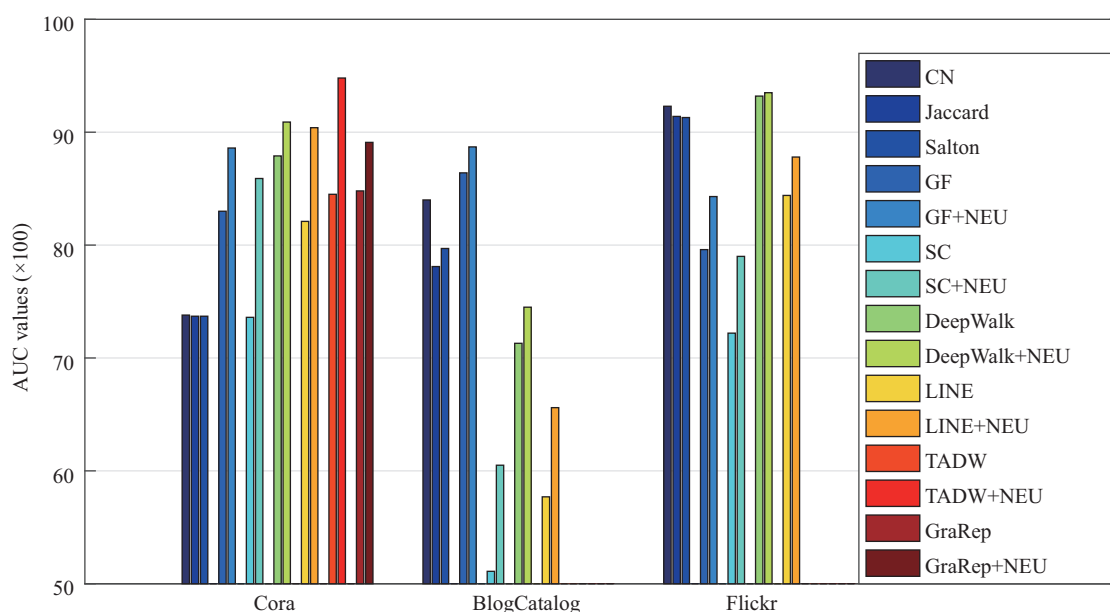


图 11 (网络版彩图) 链接预测实验结果  
Figure 11 (Color online) Link prediction results

时间. 其中 GraRep 和 node2vec 算法因其实现效率问题, 只在部分数据集上做了测试.

## 5.2 链接预测

链接预测旨在预测网络中丢失的边, 或者未来可能会出现的边. 在进行链接预测时, 需要对所有不在训练数据中的点对打分. 在表示学习中, 一般使用一对节点表示的内积或余弦相似度来计算得分. 我们一般用 AUC 值来评价链接预测任务的结果. AUC 值代表了一条未观测到的点对的得分比一条不存在的点对得分高的概率. 本文使用 Cora, BlogCatalog 和 Flickr 数据对一些网络表示学习算法进行了评测, 包括邻接矩阵分解 (GF)、谱聚类 (SC)、DeepWalk、LINE、TADW 和 GraRep. 在实验中, 移除了 Cora 数据的 20%, BlogCatalog 和 Flickr 数据的 50% 的边作为测试集, 并用剩下的边训练节点表示. 同时加入了 3 个常用的不基于表示学习的链接预测方法 Common Neighbors (CN), Jaccard Index 和 Salton Index 作为比较. 如图 11 所示, 每个数据集最左边的 3 列代表了传统链接预测方法的结果. 之后每两列分别代表了一种网络表示学习方法和其经过 NEU 更新之后的链接预测效果.

## 5.3 社区发现

社区发现问题旨在对网络中的节点进行无监督的聚类, 从而将网络中相似的节点归为同一个社区. 与节点分类任务相比, 社区发现任务最主要的不同就是社区发现任务是无监督的, 即没有任何已标定的数据. 作为一个自由度相对较高的任务, 社区发现也吸引了很多研究者在上面进行扩展研究. 在实际应用层面上, 社区发现算法可以用来为社交网络中的用户自动划分好友的分组, 为蛋白质网络中的各类蛋白质依照它们之间的联系自动分类.



## 6 总结与展望

本文介绍了现有的网络表示学习算法,并在节点分类、链接预测任务上列出了不同方法间的比较结果.大规模信息网络上的网络表示学习研究具有重要意义,现有网络表示学习方法主要依赖于静态的网络拓扑结构信息,而忽略了网络结构的动态性、网络中节点的异质性、节点拥有的大量外部信息等.在分布式表示学习的支持下,将极大推动网络表示学习方向的发展,在网络数据上实现更加广泛而深入的应用.因此,大规模复杂信息网络上的网络表示学习研究,是现代人工智能与智能信息服务的重要前沿研究课题和关键技术,具有重要的学术意义和广泛的应用价值.

网络表示学习研究虽然已经取得了丰富的成果,但是在以下方面仍然面临着巨大的挑战:

(1) 知识驱动的网络表示学习. 目前已有的网络表示学习模型没有考虑知识图谱等外部知识信息,而外部丰富的知识信息能够为网络表示学习及其后续应用提供推理能力. 因此,如何有效地将知识图谱等外部信息考虑进来,是网络表示学习面临的重要挑战.

(2) 大规模网络表示学习. 已有的网络表示学习模型仅仅能够针对小规模的网络进行有效的训练,而实际场景中的社交网络动辄存在上亿节点. 因此,如何克服真正大规模社交网络表示学习过程中,面临的表示存储、训练效率、异构信息的融合等问题,是网络表示学习迈向实际应用场景的亟待解决的关键挑战.

(3) 结合具体应用的网络表示学习. 已有的网络表示学习模型,其训练过程一般使得节点表示尽可能拟合网络的真实结构,而忽略这些节点表示后续的应用. 因此,如何能够发掘网络表示学习更多的应用场景,以及如何在网络表示学习的过程中,考虑具体的后续应用场景,有目的地改善其在具体应用场景中的表现,是网络表示学习在应用方面的重要挑战.

## 参考文献

- 1 Perozzi B, Al-Rfou R, Skiena S. Deepwalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, 2014. 701–710
- 2 Eades P, Tamassia R. Algorithms for drawing graphs: an annotated bibliography. Comput Geom, 1994, 4: 235–282
- 3 Fruchterman T M J, Reingold E M. Graph drawing by force-directed placement. Softw Pract Exper, 1991, 21: 1129–1164
- 4 Kamada T, Kawai S. An algorithm for drawing general undirected graphs. Inf Process Lett, 1989, 31: 7–15
- 5 Kobourov S G. Spring embedders and force directed graph drawing algorithms. arXiv:1201.3011
- 6 Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding. Science, 2000, 290: 2323–2326
- 7 Saul L K, Roweis S T. An introduction to locally linear embedding. 2000. <http://www.cs.toronto.edu/roweis/lle/publications.html>
- 8 Belkin M, Niyogi P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In: Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic, Vancouver, 2001. 585–591
- 9 Tang L, Liu H. Leveraging social media networks for classification. Data Min Knowl Discov, 2011, 23: 447–478
- 10 Chen M, Yang Q, Tang X. Directed graph embedding. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, 2007. 2707–2712
- 11 Tang L, Liu H. Relational learning via latent social dimensions. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, 2009. 817–826
- 12 Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. In: Proceedings of Advances in Neural Information Processing Systems, Lake Tahoe, 2013. 3111–3119
- 13 Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space. arXiv:1301.3781
- 14 Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model. In: Proceedings of International Speech Communication Association, Makuhari, 2010. 1045–1048

- 15 Tang J, Qu M, Wang M, et al. Line: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, Florence, 2015. 1067–1077
- 16 Yang C, Liu Z, Zhao D, et al. Network representation learning with rich text information. In: Proceedings of the 24th International Conference on Artificial Intelligence, Buenos Aires, 2015. 2111–2117
- 17 Cao S, Lu W, Xu Q. Grarep: learning graph representations with global structural information. In: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, Melbourne, 2015. 891–900
- 18 Yang C, Sun M S, Liu Z Y, et al. Fast network embedding enhancement via high order proximity approximation. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), Melbourne, 2017
- 19 Wang D, Cui P, Zhu W. Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, 2016. 1225–1234
- 20 Yang J, Leskovec J. Overlapping community detection at scale: a nonnegative matrix factorization approach. In: Proceedings of the 6th ACM International Conference on Web Search and Data Mining, Rome, 2013. 587–596
- 21 Ou M, Cui P, Pei J, et al. Asymmetric transitivity preserving graph embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, 2016. 1105–1114
- 22 Tu C, Wang H, Zeng X, et al. Community-enhanced network representation learning for network analysis. arXiv:1611.06645
- 23 Thomas L G, Mark S. Finding scientific topics. Proc National Acad Sci, 2004, 101: 5228–5235
- 24 Tu C C, Liu H, Liu Z Y, et al. CANE: context-aware network embedding for relation modeling. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, 2017. 1722–1731
- 25 Tu C C, Zhang W C, Liu Z Y, et al. Max-Margin DeepWalk: discriminative learning of network representation. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), New York, 2016
- 26 Li J Z, Zhu J, Zhang B. Discriminative deep random walk for network classification. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, 2016. 1004–1013
- 27 Grover A, Leskovec J. Node2vec: scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, 2016. 855–864
- 28 Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. In: Proceedings of the 5th International Conference on Learning Representations, Toulon, 2017
- 29 Yang Z, Cohen W, Salakhutdinov R. Revisiting semi-supervised learning with graph embeddings. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning, New York, 2016
- 30 Tu C C, Zhang Z Y, Liu Z Y, et al. TransNet: translation-based network representation learning for social relation extraction. In: Proceedings of International Joint Conference on Artificial Intelligence (IJCAI), Melbourne, 2017

## Network representation learning: an overview

Cunchao TU<sup>1,2,3†</sup>, Cheng YANG<sup>1,2,3†</sup>, Zhiyuan LIU<sup>1,2,3\*</sup> & Maosong SUN<sup>1,2,3</sup>

1. *Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China;*

2. *State Key Lab on Intelligent Technology and Systems, Tsinghua University, Beijing 100084, China;*

3. *National Lab for Information Science and Technology, Tsinghua University, Beijing 100084, China*

\* Corresponding author. E-mail: liuzy@tsinghua.edu.cn

† Equal contribution

**Abstract** Networks are important ways of representing objects and their relationships. A key problem in the study of networks is how to represent the network information properly. With the developments in machine learning, feature learning of network vertices has become an important area of study. Network representation learning algorithms turn network information into dense, low-dimensional real-valued vectors that can be used as inputs for existing machine learning algorithms. For example, the representation of vertices can be fed to a classifier such as a Support Vector Machine (SVM) for vertex classification. In addition, the representations can be used for visualization by taking the representations as points in a Euclidean space. The study of network representation learning has attracted the attention of many researchers. In this article, recent works on network representation learning are introduced and summarized.

**Keywords** networks, representation learning, machine learning, deep learning, neural network



**Cunchao TU** was born in 1990. He is a Ph.D. candidate and a student member of China Computer Federation. His research interests include social computing and network representation learning.



**Cheng YANG** was born in 1992. He is a Ph.D. candidate and a student member of China Computer Federation. His research interests include social computing and network representation learning.



**Zhiyuan LIU** was born in 1984. He is a Ph.D. recipient, an assistant professor and a senior member of China Computer Federation. His research interests include natural language processing, representation learning, and computational social sciences.



**Maosong SUN** was born in 1962. He is a Ph.D. recipient, professor and Ph.D. supervisor and a senior member of China Computer Federation. His research interests include natural language processing, Chinese language computing, web intelligence, and computational social sciences.