

# 网络表示学习算法综述

丁 钰 魏 浩 潘志松 刘 鑫

陆军工程大学指挥控制工程学院 南京 210000

(yuding@live.com)



**摘 要** 网络是一系列节点和边的集合,通常表示成一个包含节点和边的图。许多复杂系统都以网络的形式来表示,如社交网络、生物网络和信息网络。为了使网络数据的处理变得简单有效,针对网络中节点的表示学习成为了近年来的研究热点。网络表示学习旨在为网络中的每个节点学习一个低维稠密的表示向量,进而可将得到的向量表示运用到常见的网络分析任务中,如节点聚类、节点分类和链路预测等。然而,绝大多数真实网络节点都有丰富的属性信息,如社交网络中的用户资料和引文网络中的文本内容。网络的属性信息对网络表示具有重要的作用,当网络高度稀疏时,网络的属性信息是网络表示重要的辅助信息,有助于更好地学习网络表示。传统的邻接矩阵仅仅表示了边的信息,而无法加入节点的属性信息。因此,网络表示不仅要保存网络的结构信息,还要保存网络的属性信息。此外,大多数真实世界网络都是动态变化的,这种变化包括网络节点的增加和减少,以及网络边的新建和消失。同时,与网络结构变化相似,网络中的属性也会随着时间的推移发生变化。随着机器学习技术的发展,针对网络表示学习问题的研究成果层出不穷,文中将针对近年来的网络表示学习方法进行系统性的介绍和总结。

**关键词:**网络;网络表示学习;机器学习;网络嵌入;深度学习

中图法分类号 TP181

## Survey of Network Representation Learning

DING Yu, WEI Hao, PAN Zhi-song and LIU Xin

Institute of Command and Control Engineering, Army Engineering University of PLA, Nanjing 210000, China

**Abstract** A network is a collection of nodes and edges, usually is represented as a graph. Many complex systems take the form of networks, such as social networks, biological networks, and information networks. In order to make network data processing simple and effective, the representation learning for nodes in the network has become a research hotspot in recent years. Network representation learning is designed to learn a low-dimensional dense representation vector for each node in the network that can advance various learning tasks in the network analysis area such as node classification, network clustering, and link prediction. However, most of previous works have been designed only for plain networks and ignore the node attributes. When the network is high sparsity, attributes can be the very useful complementary content to help learn better representations. Therefore, the network embedding should not only preserve the structural information, but also preserve the attribute information. In addition, in practical applications, many networks are dynamic and evolve over time with the addition, changing and deletion of nodes. Meanwhile, similar as network structure, node attributes also change naturally over time. With the development of machine learning, studies on the network representation learning emerge one after another. In this paper, we will systematically introduce and summarize the network representation learning methods in recent years.

**Keywords** Networks, Network representation learning, Machine learning, Network embedding, Deep learning

## 1 引言

近年来,随着社会的飞速发展,各领域的的数据量不断增加,特别是大规模的社交网络不断涌现,如微博、微信、Facebook、Twitter等。

面对大规模网络,传统的网络建模方法不再适用。在大规模问题上高效地进行网络分析任务,例如节点聚类、节点分类、链路预测、重要节点发现等,成为了当务之急。这些网络分析任务的关键问题是:如何利用网络中的结构信息和属性

信息来有效地对网络中的节点进行表示。传统的邻接矩阵仅仅表示了边的信息,而无法加入节点本身的属性。在当前的很多应用中,我们不仅需要表示图的结构属性,还需要融合节点属性。网络表征学习的目的有两个:1)学习一个低维的向量来表示网络的节点,将节点映射到相应的向量空间,这种网络表征学习就是关于图结构的网络表征学习;2)网络表示不仅要表征网络结构,也要表征节点本身的属性信息。大规模网络的形成,使得传统的网络表示方法遇到了瓶颈。

随着机器学习技术的快速发展,作为机器学习算法的输

到稿日期:2019-03-06 返修日期:2019-06-25 本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。

基金项目:国家自然科学基金(61473149)

This work was supported by the National Natural Science Foundation of China(61473149).

通信作者:魏浩(594386708@qq.com)

入,网络节点的低维表示学习变得越来越重要。网络表示学习已经成为数据挖掘的研究热点,大量的研究成果层出不穷。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

网络表示学习(Network Representation Learning),又名网络嵌入(Network Embedding)、图嵌入(Graph Embedding)。网络表示是衔接网络原始数据和网络应用任务的桥梁,旨在将网络中的节点表示成低维、实值、稠密的向量形式,使得到的向量形式可以在向量空间中具有表示以及推理的能力,将得到的向量表示运用到常见的社交网络应用中。

网络  $G$  定义为:  $G=(V,E)$ 。其中,  $V=\{v_1, v_1, \dots, v_n\}$  表示网络中的节点集合,  $E$  表示节点的边集合,  $n$  为网络中的节点数量。网络表示学习的目标是为网络中的每个节点  $v$  学习一个低维稠密的实数向量  $x_v \in \mathbb{R}^d$ , 其中  $d \ll N$ 。网络表示学习的过程可以是无监督或者半监督的。

## 3 基于结构的网络表示学习算法

基于结构的网络表示学习旨在使学习的向量保留网络的结构信息。网络的结构信息包括网络节点的邻居结构、高阶相似性、网络社团结构等。早期的网络表示学习算法大多基于谱聚类。比较有代表性的工作包括: PCA<sup>[1]</sup>、局部线性表示<sup>[2-3]</sup>、Laplace 特征表<sup>[4-5]</sup>、有向图表示<sup>[6]</sup>等。PCA 采用线性降维技术得到网络表示;局部线性表示通过其邻居节点表示的线性加权组合得到;Laplace 特征表假设节点与其邻居节点的距离很近;有向图表示通过给不同节点的损失函数赋予不同的权重,来对 Laplace 特征表进行进一步扩展。基于谱聚类的方法由于求取关系矩阵的特征向量的时间是非线性的,并且需要将关系矩阵存于内存中,因此时间和空间复杂度较高,不适用于大规模网络。

随着自然语言处理中词嵌入技术 word2vec<sup>[7-9]</sup> 的提出, DeepWalk<sup>[10]</sup> 和 Node2vec<sup>[11]</sup> 将节点视为单词并生成随机游走节点序列作为句子。然后,可以将诸如 Skip-gram<sup>[12]</sup> 之类的神经语言模型应用于这些随机游走序列,以获得网络嵌入。Node2vec 通过两个参数  $p$  和  $q$  来控制随机游走的跳转概率。假设上一步游走从节点  $v_{i-1}$  到节点  $v_i$ , 则到下一步节点  $v_{i+1}$  的未归一化的跳转概率  $\pi_{v_i v_{i+1}} = \alpha_{pq}(v_{i-1} v_{i+1})$  为:

$$\alpha_{pq}(v_{i-1} v_{i+1}) = \begin{cases} \frac{1}{p}, & \text{if } d_{v_{i-1} v_{i+1}} = 0 \\ 1, & \text{if } d_{v_{i-1} v_{i+1}} = 1 \\ \frac{1}{q}, & \text{if } d_{v_{i-1} v_{i+1}} = 2 \end{cases}$$

其中,  $d_{v_{i-1} v_{i+1}}$  表示节点  $v_{i-1}$  和节点  $v_{i+1}$  之间的最短距离。对于节点序列中的节点  $v_i$ , 假设窗口为  $w$ , 则损失函数可以由下式表示:

$$\max_{\phi} \Pr(\{v_{i-w}, \dots, v_{i+w}\} / v_i | \phi(v_i)) = \prod_{j=i-w, j \neq i}^{i+w} \Pr(v_j | \phi(v_i))$$

其中,  $\phi(v_i)$  为节点  $v_i$  的表示向量,  $\{v_{i-w}, \dots, v_{i+w}\} / v_i$  表示节点  $v_i$  的上下文节点。Node2vec 生成随机游走序列和优化 Skip-gram 模型的过程都是高效且可以并行化的。Node2vec 是 DeepWalk 的扩展, 它引入了一个偏向的随机游走, 该游走

结合了 BFS 风格和 DFS 风格的邻域探索。

为了克服邻接矩阵的稀疏性, LINE<sup>[13]</sup> 同时考虑了网络节点的一阶邻近度(First-order Proximity)和二阶邻近度(Second-order Proximity)。一阶邻近度指两个直接相连的节点表示应该尽可能地相近。一阶邻近度通过两个直接相连节点表示的联合概率来表示:

$$p_1(v_i, v_j) = \frac{1}{1 + \exp(-\vec{u}_i^T \cdot \vec{u}_j)}$$

二阶邻近度指对于两个不直接相连的节点, 如果它们的邻居相似, 则其学习到的节点表示也应该相近。二阶邻近度用两个节点表示的条件概率表示:

$$p_1(v_i | v_j) = \frac{\exp(\vec{u}_j^T \cdot \vec{u}_i)}{\sum_{k=1}^{|V|} \exp(\vec{u}_k^T \cdot \vec{u}_i)}$$

在 LINE 中, 只有距离给定节点最多两跳的节点才被视为给定节点的相邻节点。此外, 与 DeepWalk 中使用的分层 softmax 相比, LINE 使用负采样来优化 Skip-gram 模型。

考虑到 LINE 只保留了一阶和二阶邻近度, GraRep<sup>[14]</sup> 通过一个特殊的关系矩阵保留了  $k$ -step( $k > 2$ ) 阶邻近度。对于给定的归一化后的邻接矩阵  $A$ , 关系矩阵的定义为:

$$A^k = \underbrace{A \cdots A}_k$$

$A^k$  矩阵中的每个元素  $A_{ij}^k$  表示节点  $i$  通过  $k$  步随机游走抵达节点  $j$  的概率。节点  $i$  的  $k$  步损失函数的定义为:

$$L_k(i) = \sum_j ((j|i) \log \sigma(\vec{u}_j^T \vec{u}_i)) + \lambda \mathbb{E}_{j' \sim p_k(V)} [\log \sigma(-\vec{u}_i^T \vec{u}_{j'})]$$

其中,  $\sigma(x) = (1 + e^{-x})^{-1}$ ,  $p_k(V)$  表示网络中节点的分布,  $j'$  表示通过负采样获得的节点。GraRep 对不同的  $k$  值分别进行了计算, 然后将对应的网络表示拼接起来, 形成表达能力更强的节点表示。GraRep 的主要缺点为关系矩阵  $A^k$  的计算效率很低, 不适用于大规模网络。

现有的保存网络高阶邻近度的网络表示学习方法都只能保存固定阶数的邻近度, 并且不能同时保证建模的准确性和计算效率。鉴于此, AROPE<sup>[15]</sup> 在 SVD 框架的基础上学习网络高阶邻近度。AROPE 通过理论证明揭示了网络节点各阶邻近度的内在关系, 并提出了一种基于特征分解的网络表示学习方法。AROPE 学到的网络表示可在各阶邻近度中任意转换, 并且转换的效率很高。

其他的基于网络结构的网络表示学习方法还包括: HOPE<sup>[16]</sup> 使用 JDGSVD 算法对网络的非对称关系矩阵进行降维来学习节点的低维表示, 其可以保存网络中的非对称信息; NetHiex<sup>[17]</sup> 提出了一种全新的网络表示学习方法, NetHiex 协同训练节点表示和类别表示, 节点表示由多个不同粒度的类别表示组成。

为了保存网络中重要的全局社团结构信息, 一些基于网络社团结构的网络表示方法相继被提出。Wang 等<sup>[18]</sup> 基于矩阵分解技术提出了 M-NMF, 该方法将网络的社团结构结合到网络表示中。M-NMF 利用节点表示和社团间的共识关系, 将基于模块度的社团检测模型和基于 NMF 的网络表示模型在统一的框架中进行优化, 使节点表示能够保留网络的社团结构信息。Cavallari 等<sup>[19]</sup> 发现对网络中的社团进行表示不仅对社团级别的应用有益, 而且有利于社团检测和节点分类。在此基础上, 他们提出了一种新的网络表示学习模型 ComE。在 ComE 中, 每个社区被表示成一个高维高斯分布

来描述该社区中的网络节点的分布情况。通过对模型的精心设计, ComE 可以同时学习优化社团检测、网络节点表示和社团表示。然而, ComE 仅仅尝试求解最能够拟合节点表示的高维高斯分布, 而没有显式考虑社区的网络结构信息。CNRL<sup>[20]</sup>扩展了网络结构和文本之间的类比关系, 采用主题建模的方法来同时进行重叠社区检测和网络表示学习。在 CNRL 中, CNRL 通过随机游走生成节点序列, 然后同时利用每个节点及其分配的社区来预测序列中的上下文节点。这样, 在进行网络表示学习的过程中, 节点的社团分布也会同步进行更新。然而, 该模型建立的主题和社团之间的相关性没有理论依据, 缺乏基于矩阵分解模型的直观可解释性。

现实中网络的结构往往复杂多样, 网络表示学习通常挖掘网络的邻域结构、节点高阶相似性和网络的社团结构。基于网络结构的表示学习旨在使学习的节点表示保存网络的结构信息, 例如, 相同社团的节点的表示应该相似, 邻域相同的节点的表示也应该相似。网络的表示由于保存了节点的结构信息, 因此可以完成各种网络分析任务, 如节点分类、链接预测、重要节点发现等。

#### 4 结合属性的网络表示学习算法

现实生活中的很多网络节点都伴随着丰富的额外信息, 如节点标签、节点属性、节点的语义描述等。如何将这些属性信息与网络拓扑结合起来学习网络的表示, 引起了很多学者的兴趣。

Le 等<sup>[21]</sup>针对文本网络提出了一个生成模型对网络进行学习, 文本网络中的节点可以是超链接网页、社交网络中的用户、学术出版物的引文等, 每个节点均有相应的文本作为属性信息。该模型同时考虑单词与文本以及文本与文本之间的关系, 并根据主题模型 RTM<sup>[22]</sup>为每个节点学习一个可解释性的表示。

TADW<sup>[23]</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。TADW 首先证明了 DeepWalk 方法等价于对关系矩阵  $M$  进行分解。矩阵  $M$  的元素为:

$$M_{ij} = \log([e_i(A + A^2 + \dots + A^t)]_j / t)$$

其中,  $A$  为邻接矩阵;  $t$  代表随机游走的步数;  $e_i$  向量中除了第  $i$  个元素为 1 外, 其余都为 0。根据 Natarajan 等<sup>[24]</sup>的工作, TADW 结合节点的文本特征  $T$  建立如下目标函数:

$$\min_{W, H} \|M - W^T H T\|_F^2 + \frac{\lambda}{2} (\|W\|_F^2 + \|H\|_F^2)$$

最后, 模型将  $W$  和  $HT$  拼接后作为最后的节点表示。TADW 面临着与谱聚类方法同样的问题, 即对关联矩阵  $M$  的计算和存储效率较低, 难以适用于大规模的社交网络。

随着基于 Skip-gram 模型的 DeepWalk 方法被提出, 一些基于 DeepWalk 的方法也相继被提出。Ahmed 等<sup>[25-26]</sup>提出了一种基于随机游走的属性网络表示方法, 首先利用节点的属性信息将网络节点映射为一系列节点类型, 然后利用节点类型进行属性随机游走, 生成节点类型序列, 最后使用 Skip-gram 模型学习网络表示。该方法为每个节点类型学习一个表示向量, 因此相同类型的节点的表示向量也相同。Nguyen 等<sup>[27]</sup>在 DeepWalk 的基础上提出了另一种随机游走方法, 即 BiasedWalk。该方法可以保存网络的高阶相似性, 并可以同时扮演深度优先搜索 (Depth First Search, DFS) 和宽度优先

搜索 (Breadth First Search, BFS) 两种角色。另外, BiasedWalk 可以控制随机游走节点与起始采样点的距离, 这在网络链路预测任务中有着很重要的作用。

真实世界中的网络节点总是因不同方面的特点与其他节点进行交互。例如, 社交网络中的用户往往因为不同的爱好, 与不同的人成为朋友。为了学习到同一个节点对不同邻居展现的不同方面, CANE<sup>[28]</sup>利用节点的文本信息, 并根据不同的邻居学习网络表示。CANE 假设文本表示向量的生成过程与邻居相关, 并利用 CNN 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表示向量。

网络中的部分节点往往还有标签信息, 为了更好地利用节点的属性信息和标签信息, Pan 等<sup>[29]</sup>提出了一个耦合深度模型 TriDNR。假设网络的节点集合为  $\{v_i\}_{i=1, \dots, N}$ , 每个节点有一个单词集合  $\{w_i\}$  作为节点属性, 网络中的部分节点具有标签  $\{c_i\}$ 。TriDNR 采用随机游走生成节点序列  $S$ 。为了挖掘节点的结构信息、属性信息、标签信息, TriDNR 最大化如下函数:

$$L = (1 - \alpha) \sum_{i=1}^N \sum_{S \in S} \sum_{-b \leq j \leq b, j \neq 0} \log P(v_{i+j} | v_i) + \alpha \sum_{i=1}^N \sum_{-b \leq j \leq b} \log P(w_j | v_i) + \alpha \sum_{i=1}^L \sum_{-b \leq j \leq b} \log P(w_j | c_i)$$

其中,  $b$  为节点序列选取的上下文窗口大小。上式中的第一项与 DeepWalk 相似, 是对网络结构的建模; 第二项是对节点与属性关系的建模; 第三项是对节点与标签的建模。因此, 学习的网络表示同时保存了网络的结构、属性、标签信息。

网络表示学习大多是基于无监督的, 针对节点分类等机器学习任务, 其缺少区分性。受 MedLDA<sup>[30]</sup>的启发, Tu 等<sup>[31]</sup>针对这个问题提出了 MMDW 方法来学习有区分性的网络表示。MMDW 同时学习一个基于 DeepWalk 的网络表示模型和一个基于 SVM<sup>[32]</sup>的最大间隔分类器。MMDW 的优化目标如下:

$$\min_{X, Y, W, \xi} L = \min_{X, Y, W, \xi} L_{DW} + \frac{1}{2} \|W\|_2^2 + C \sum_{i=1}^T \xi_i$$

$$\text{s. t. } w_i^T x_i - w_j^T x_j \geq e_i^j - \xi_i, \forall i, j$$

其中,  $W = [w_1, \dots, w_m]^T$  为 SVM 的权重矩阵,  $\xi = [\xi_1, \dots, \xi_T]$  是控制集中分类错误的松弛变量; 当  $i = j$  时,  $e_i^j = 1$ , 否则  $e_i^j = 0$ ;  $\min_{X, Y, W, \xi} L_{DW}$  表示基于 DeepWalk 的矩阵分解模型的目标函数, 其表达式如下:

$$L_{DW} = \|M - (X^T Y)\|_2^2 + \frac{\lambda}{2} (\|X\|_2^2 + \|Y\|_2^2)$$

其中, 超参数  $\lambda$  控制正则项的权重。关系矩阵  $M$  采用 TADW 中类似的设定, 取  $M = (A + A^2)/2$ 。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, 从而提升表示向量的区分能力。

网络中的节点往往只有部分标签, 并且节点的标签信息可能包含很多噪音。为了解决上述问题, Huang 等<sup>[33]</sup>提出了 LANE 模型, 该模型主要基于谱聚类<sup>[34]</sup>技术。LANE 首先使用余弦相似度构造节点属性、网络结构和标签的相应 Affinity 矩阵, 然后根据拉普拉斯矩阵学习到 3 个不同的表示, 最后将所有这些表示映射到同一共同空间作为最后的节点表示。该算法的核心在于将 3 个独立学习所得的矩阵空间进行融合, 然后通过公式更新来优化初始矩阵, 从而达到最终的网络向

量空间的映射。为了建模网络中属性和结构的非线性关系, STNE<sup>[35]</sup>采用 LSTM<sup>[36]</sup>对网络节点序列进行编码和解码,通过学习节点的属性序列到节点标识序列的映射,可以将网络的结构信息和属性信息融入节点表示。

网络的拓扑结构是网络最显著的特征,但现实网络的节点往往包含了丰富的属性特征,这些属性通常驱动着网络的形成和发展。本小节回顾了具有代表性的属性网络表示学习方法,并证明了网络属性对节点表示的重要性。属性网络表示学习方法不仅要保存网络的结构信息,还要保存网络的属性信息。原始网络的结构信息和属性信息的差异性和异质性,使得有效地将原始网络的结构信息和属性信息融入节点的低维表示中成为了属性网络表示学习最大的挑战。

## 5 基于深度神经网络的表示学习算法

网络的结构和属性信息都存在高度的非线性关系,捕获它们的非线性关系对于网络表示非常重要,浅层模型不能学到网络的非线性关系。随着深度模型在各领域的成功运用,如何将深度模型应用到网络数据成为了捕获网络中非线性关系的关键问题。

SDNE<sup>[37]</sup>使用深度神经网络对节点间的非线性进行建模。SDNE使用深度自编码器(Deep Autoencoder)来保留节点的邻居结构。对于给定的邻接矩阵中节点  $i$  对应的行  $x_i$ ,深度自编码器的每个隐层输出如下:

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

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

解码器的输出  $\hat{x}_i$  通过计算解码器的相反过程来获得。为了给非零元素的重构误差施加更多的惩罚,SDNE引入惩罚向量  $b_i = \{b_{ij}\}_{j=1}^n$ 。如果节点  $i$  与节点  $j$  之间有一条边,则对应的  $b_{ij}$  将大于 1。网络重构损失函数如下:

$$L_{2nd} = \sum_i \|\hat{x}_i - x_i \odot b_i\|^2$$

上式保留了重构网络保存节点的二阶相似性。为了保存节点的一阶相似性,SDNE采用了 Laplacian eigenmaps。

为了更好地保存网络的结构信息和节点的属性信息,Zhang 等<sup>[38]</sup>提出了基于深度神经网络的属性网络表示学习模型 ANRL。该模型使用一个邻居增强的深度自编码来建模节点的属性信息,网络的结构信息则通过带属性感知的 Skip-gram 进行建模。ANRL 中自编码器的重构损失函数如下:

$$L_{ae} = \sum_{i=1}^n \|\hat{x}_i - T(v_i)\|_2^2$$

其中,  $\hat{x}_i$  为解码器的重构输出,  $T(v_i)$  表示节点  $v_i$  邻居的一种变换。可以看出,自编码器的重构目标为节点  $v_i$  的邻居信息。 $T(v_i)$  可以采用如下两种方式得到:

$$T(v_i) = \frac{1}{|N(i)|} \sum_{j \in N(i)} w_{ij} x_j$$

或者

$$T(v_i) = [\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_m]$$

其中,  $N(i)$  表示节点  $v_i$  的邻居节点,  $\tilde{x}_k$  表示其相关邻居属性向量在第  $k$  维的中值,具体如下:

$$\tilde{x}_k = \text{Median}(w_{i1} x_{1k}, w_{i2} x_{2k}, \dots, w_{i|N(i)|} x_{|N(i)|k})$$

其中,  $\text{Median}(\cdot)$  返回输入的中值。通过该自编码器,模型捕获了节点的属性信息。网络的结构信息则通过如下带属性感知的 Skip-gram 保存:

$$L_{ag} = - \sum_{i=1}^n \sum_{c \in C} \sum_{b \leq j \leq b, b \neq 0} \log p(v_{i+j} | x_i)$$

其中,  $b$  为节点序列选取的上下文窗口大小。通过联合训练  $L_{ae}$  和  $L_{ag}$ ,模型可以同时保存网络的结构信息和节点的属性信息。

为了将 CNN 应用于网络数据学习网络表示, Kipf 等<sup>[39]</sup>提出了一种图卷积神经网络模型来表示属性网络,但该模型为一种半监督方法,不能在无监督条件下建模。同年, Kipf 等<sup>[40]</sup>又提出了一种无监督模型 VGAE, VGAE 在 VAE<sup>[41]</sup>的基础上发展而来,可以方便地利用节点属性对网络进行表示。SiNE<sup>[42]</sup>是一种符号网络表示学习方法。对于给定的节点三元组  $(v_i, v_j, v_k)$ ,  $e_{ij} = 1$  表示节点  $v_i$  和  $v_j$  之间存在正边,  $e_{ik} = -1$  表示节点  $v_i$  和  $v_k$  之间存在负边;  $x_i, x_j, x_k$  分别表示对应节点的表示向量,则  $x_i$  与  $x_j$  的相似度  $f(x_i, x_j)$  应该大于  $x_i$  与  $x_k$  的相似度  $f(x_i, x_k)$ 。

现有网络表示学习方法往往将节点间的相似性定义为有边的节点或者有相同邻居的节点,然而,分布在网络中不同位置的两个节点也有可能扮演相同的角色。DRNE<sup>[43]</sup>采用 LSTM,通过递归的方式聚合邻域节点的表示来得到节点的表示。其损失函数为:

$$L_1 = \sum_{v \in V} \|\mathbf{X}_v - \text{Agg}(\{\mathbf{X}_u | u \in N(v)\})\|_F^2$$

其中,  $\text{Agg}$  表示某个聚合函数,  $\mathbf{X}_v$  表示节点  $v$  的表示,  $N(v)$  表示节点  $v$  的邻居。在一个递归步骤中,学习的节点表示保存了邻居节点的结构信息,通过迭代更新,使节点表示保存网络的全局信息。为了保存网络的非线性关系,DRNE 采用 LSTM 作为聚合函数。为了避免所有节点表示为 0 的情况,DRNE 利用节点的度作为引导信息设计了如下的约束,使节点表示与节点的度相似:

$$L_{reg} = \sum_{v \in V} \|\log(d_v + 1) - \text{MLP}(\text{Agg}(\{\mathbf{X}_u | u \in N(v)\}))\|_F^2$$

其中,  $d_v$  表示节点  $v$  的度,  $\text{MLP}$  为一个多层感知器。DRNE 的目标函数如下:

$$L = L_1 + \lambda L_{reg}$$

其中,  $\lambda$  为权重参数。图 1 展示了 DRNE 的基本框架。

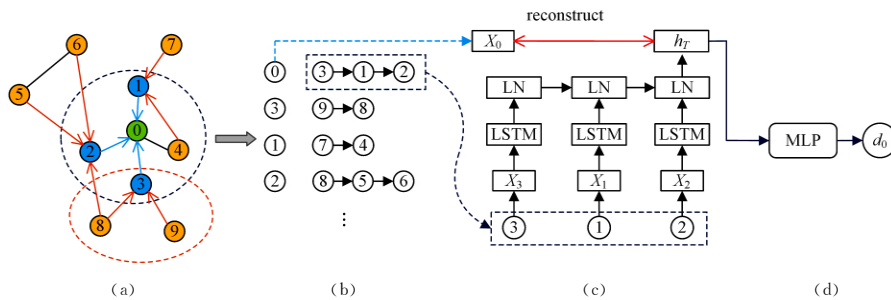


图 1 DRNE 的框架<sup>[43]</sup>

Fig. 1 Framework of DRNE<sup>[43]</sup>



最近出现的基于深度神经网络的网络表示学习方法还包括:DVNE<sup>[44]</sup>在 Wasserstein 空间中为网络中的节点学习了一个高斯分布作为节点的表示,该表示可以保存网络的结构信息以及建模节点的形成和演变的不确定性;GAT<sup>[45]</sup>将自注意力机制应用到图数据中来学习网络节点的表示,节点在更新自身特征时能够兼顾其他所有邻居节点的特征;GAT 不需要预先知道整个网络的结构,可以对同一个邻域的节点隐式地指定不同的权重,有利于模型的可解释性;GraphSAGE<sup>[46]</sup>提出了一种基于邻居特征聚集的网络表示学习方法,其通过聚集采样到的邻居特征更新当前节点的特征,GraphSAGE 可以为新增节点快速生成节点表示,而无需额外的训练过程。

基于深度神经网络的网络表示学习方法主要对节点表示间的非线性进行建模,能够捕获网络结构和属性中的非线性

关系。但是,这些方法大多需要很长的计算时间,往往很难处理现实生活中的大规模网络。

## 6 动态网络的表示学习算法

许多现实世界的网络,特别是社交网络,并不是静态的,而是不断动态发展的。因此,许多网络挖掘任务的结果随着网络的变化需要不断更新。现有的针对动态网络表示的工作大多是将静态网络的方法应用在动态网络的每个时间片上<sup>[47-48]</sup>,而直接将现有静态表示算法独立应用于每个网络快照上的表现往往不尽如人意。

DANE<sup>[49]</sup>提供了一种用于动态网络表示学习的在线更新方法,其运用矩阵扰动理论<sup>[50]</sup>来使网络表示以在线的方式保持更新。DANE 算法的流程如图 2 所示。

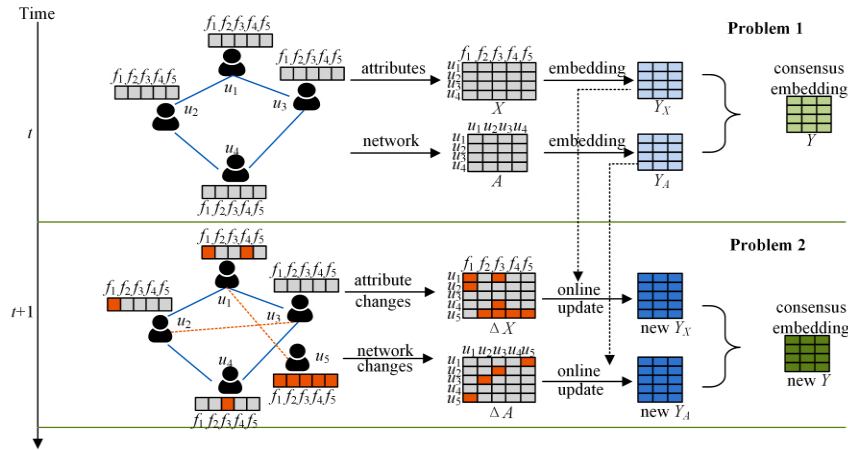


图 2 DANE 框架图解<sup>[49]</sup>

Fig. 2 Illustration of framework-DANE<sup>[49]</sup>

在  $t$  时刻,DANE 首先根据基于谱聚类的网络表示方法<sup>[4,51]</sup>针对网络的邻接矩阵  $A$  和属性矩阵  $X$  分别学习网络的表示  $Y_A$  和  $Y_X$ 。然后,为了捕获它们之间的相互依赖性,DANE 利用文献<sup>[51]</sup>中的方法最大化  $Y_A$  和  $Y_X$  的相关性来得到最后的网络表示  $Y$ 。在  $t+1$  时刻,DANE 根据矩阵理论来在线更新  $Y_A$  和  $Y_X$ ,然后得到新的网络表示  $Y$ 。但是,DANE 面临着与谱聚类方法同样的问题,即求取关系矩阵的特征向量的时间是非线性的,难以适用于大规模的社交网络。

Goyal 等<sup>[52]</sup>提出了一种高效的动态网络表示学习方法 DynGEM。DynGEM 的主要优点包括:1)学习的网络表示随着时间的变化变得稳定;2)可以处理不断增长的动态网络;3)比在每个时间片上使用静态网络表示方法更高效。在每个时间片上,DynGEM 使用深度自编码器来学习网络的表示。DynGEM 算法的流程如图 3 所示,在  $t$  时刻,DynGEM 利用 SDNE<sup>[37]</sup>的方法,通过一个深度自编码器学习网络表示。在  $t+1$  时刻,DynGEM 采用 Net2WiderNet 和 Net2DeeperNet<sup>[53]</sup>方法在线更新自编码器的每个权重变量。另外,DynGEM 通过一种启发式方法 PropSize 来增加自编码器的深度和宽度以适应网络规模的增长。PropSize 的具体流程为:对于自编码器,从输入层开始,连续的两层需要满足如下约束:

$$size(l_{k+1}) \geq \rho \times size(l_k)$$

其中,  $0 < \rho < 1$ 。如果存在任意的连续两层  $(l_k, l_{k+1})$  不满足上述约束,则增加  $l_{k+1}$  的宽度,直到满足条件。编码器的输出

层宽度始终保持不变。如果编码器的倒数第二层和编码器的输出层不满足 PropSize 条件,则在它们之间增加编码器的层数,直到条件满足为止。对于解码器,从输出层开始也使用同样的方法来增加解码器的宽度和深度。DynGEM 通过利用前一时刻的信息,使得当前时刻只需要微调模型参数,从而加速了模型的训练过程,并且使学习的网络表示在连续的两个时刻保持稳定。

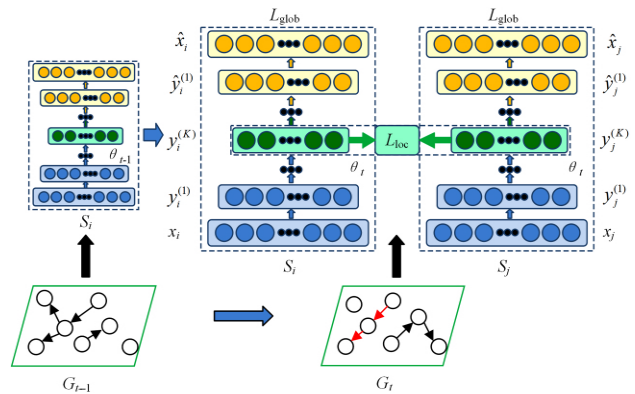


图 3 DynGEM 动态图嵌入模型<sup>[52]</sup>

Fig. 3 DynGEM dynamic graph embedding model<sup>[52]</sup>

随着链接和节点的添加、删除和更改,动态网络不断发展,不同的边在网络中建立的时间不同,网络中边建立的时间信息对网络表示也会产生一定的影响。例如,在社交网络中,

一个用户多年前的小学同学与现在的研究生同学对该用户的影响不同,刚认识的研究生同学往往对他的影响更大;并且,该用户的小学同学与其研究生同学认识的概率也不大。之前很多工作都忽略了动态网络中边建立的时间信息。为了解决上述问题,Nguyen 等<sup>[54]</sup>提出了一个结合动态网络中时间信息的网络表示学习框架 CTDNE。该方法基于 DeepWalk 设计了一种时间随机游走方法。随机游走选择初始边的方法有两种。

#### (1) 采用指数分布选择

$$P_r(e) = \frac{\exp[T(e) - t_{\min}]}{\sum_{e' \in E_T} \exp[T(e') - t_{\min}]}$$

其中,  $T(e)$  表示边  $e$  建立的时间,  $E_T$  表示网络中边的集合,  $t_{\min}$  表示边建立的最小时间。

#### (2) 采用线性分布选择

$$P_r(e) = \frac{\eta(e)}{\sum_{e' \in E_T} \eta(e')}$$

其中,  $\eta: E_T \rightarrow \mathbb{Z}^+$  为一个将边按时间升序排列的函数。两种选择方法都可以使新建立的边有更大概率被选择为随机游走的初始边。在 CTDNE 的时间随机游走中,当前待游走的边建立的时间必须比上一步游走的边建立的时间晚。时间随机游走边的选择也有两种方法。

#### (1) 采用指数分布游走

$$P_r(w) = \frac{\exp[T(w) - T(v)]}{\sum_{w' \in \Gamma_i(v)} \exp[T(w') - T(v)]}$$

其中,  $\Gamma_i(v)$  表示连接节点  $v$  的那些建立时间晚于上一步边的边集合。可以看出,该游走方法更倾向于选择建立时间晚的边。

#### (2) 采用线性分布游走

$$P_r(w) = \frac{\delta(w)}{\sum_{w' \in \Gamma_i(v)} \delta(w')}$$

其中,  $\delta$  为将边集合的时间按照降序排列的函数。当采样到了节点序列后,依然采用 Skip-gram 建模节点序列来学习节点表示。

最近,又出现了很多动态网络表示方法。SAGE<sup>[46]</sup>提出了一种归纳学习方法来保留动态网络中的属性信息。由于模型中的参数在训练结束后固定不变,因此模型的表达能力受到了很大的限制。DynamicTriad<sup>[55]</sup>为一个新的动态网络表示学习方法,其引入 Triad 结构——3 个顶点组成的一个网络基本单元,来作为网络动态变化的基本机制,从而使模型能够捕捉网络的动态性,并在不同的时间点学习每个顶点的向量表示。DynamicTriad 可以在保持网络结构性质的同时,将网络的时间变化特征嵌入表示向量中。DNE<sup>[56]</sup>是一种高效且稳定的动态网络表示方法。其基于 Skip-gram 模型,将动态网络表示分解成两个任务:计算新节点的表示,调整受到很大影响的原始节点的表示。DNE 通过平滑正则化,可以保证网络表示的稳定性。DHPE<sup>[57]</sup>重点挖掘动态网络演变的模式,并预测下一时刻两个节点是否存在链接。但该方法无法处理网络节点增加的问题。

本节介绍了几种典型的动态网络表示学习算法,这些方法都尝试表示网络的演变特征,但对现实网络的高度动态性

建模不够,缺乏对网络动态变化本质特征的挖掘。现有的关于动态网络的表示学习算法还很少,而动态网络在现实应用中变得越来越重要,因此对动态网络表示学习方法进行研究将有非常重要的现实意义。

## 7 网络表示学习面临的挑战

已有方法还存在如下几个方面的挑战。

(1) 属性信息与结构信息的融合。现实世界中的大多数网络的节点都具有属性信息,例如社交网络中用户的年龄、学历、工作、爱好、地址等,这些属性可以辅助我们对网络进行表示,缓解网络结构稀疏带来的问题。现有方法几乎都假设网络结构信息和属性信息对网络的描述存在一致性。但这种一致性的程度仍然是未知的,如何灵活融合两种信息到同一空间对节点进行更好的表示,目前的方法并没有很好地解决这个问题。

(2) 数据缺失与噪音。在真实的网络数据中,由于采集的不完整或者记录的错误,网络的结构信息与节点的属性信息都有可能不完整,并且可能存在很多噪音,这进一步加剧了网络表示学习的问题。当很多节点缺失属性数据时,很多基于矩阵分解的属性网络表示学习方法很难被直接使用。鉴于大多真实网络数据都存在这个问题,现有的很多方法不能直接应用到这些数据中。

(3) 计算效率。随着网络的规模越来越大,传统的网络表示学习方法越来越难以适应网络规模增长的速度。现有的网络表示学习方法,要么是基于谱聚类的方法,它们需要进行特征分解,每次迭代的时间复杂度为  $O(n^3)$  ( $n$  为网络中的节点数);要么基于深度神经网络,使用梯度下降算法,其收敛速度也很慢。如何解决现有的动辄上亿节点的网络面临的训练效率和表示存储等问题,是网络表示学习亟待解决的问题。

(4) 社团结构。真实网络中一般都存在社团结构,社团内的节点连接紧密,不同社团间的节点连接稀疏。网络中同一社团中的节点一般拥有相同的属性信息。例如在社交网络中,同一所学校的用户更倾向于形成社团。因此,网络表示学习需要学习到有区分性的节点表示,通过节点表示能够明显区分不同社团中的节点。如何学到具有区分性的节点表示,也是网络表示学习需要解决的问题。

(5) 动态网络。现实网络往往是动态变化的,这种变化主要包括:1)网络中节点的增加与减少;2)网络中边的增加与减少;3)网络节点属性的变化。现有的针对动态网络的表示方法大多是在每个时间片上运用静态网络的表示方法,这种方法由于在每个时间片上都要重新训练模型,因此会花费很多时间。另外,这些方法都尝试去表示网络的演变特征,但对现实网络的高度动态性建模还不够,缺乏对网络动态变化的本质特征的挖掘。

结束语 现实世界中的许多复杂系统都采用的是网络的形式表示。随着深度学习的快速发展,对复杂网络做出有效的表示是今后学术研究的热门话题。近年来,网络表示学习问题吸引了大量研究者的目光。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,但仍存在一些重要问题亟待解决。

(1)结合外部知识的网络表示学习。目前的网络表示学习模型仅仅保存网络自身的信息,忽略了知识图谱等外部知识信息,而这些外部信息往往有助于更好地建模网络,并可对后续的网络分析任务进行解释及推理。因此,如何有效地将这些外部知识完美地融合到网络表示是目前面临的重要问题。

(2)针对具体应用的网络表示学习。现有的网络表示学习方法大多是尽可能地保存网络的所有信息,而后将网络表示应用到后续的所有网络分析任务中。这种网络表示学习没有针对具体的应用特点设计更加有效合理的模型。因此,如何针对不同的应用场景设计更加合理的网络表示学习模型,是网络表示学习在应用中需要解决的重要问题。

(3)大规模网络表示学习。近年来,各领域的数据量不断增大,特别是大规模的社交网络不断出现。目前的网络表示学习方法为了追求表示的精确性,模型往往设计得很复杂,这些方法一般只能对小规模网络进行表示学习,而不能处理现实生活中上亿节点的大规模网络。如何设计有效的大规模网络的表示学习模型,已经成为网络表示学习迫切的现实需求。

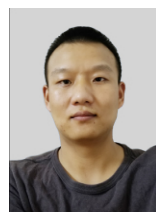
### 参 考 文 献

- [1] JOLLIFFE I T. Principal Component Analysis[J]. Journal of Marketing Research, 2002, 25(4): 513.
- [2] ROWEIS S T, SAUL L K. Nonlinear dimensionality reduction by locally linear embedding[J]. Science, 2000, 290: 2323-2326.
- [3] SAUL L K, ROWEIS S T. An introduction to locally linear embedding[J]. Journal of Machine Learning Research, 2008, 7.
- [4] BELKIN M, NIYOGI P. Laplacian eigenmaps and spectral techniques for embedding and clustering[C] // Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic. Vancouver, 2001: 585-591.
- [5] TANG L, LIU H. Leveraging social media networks for classification[J]. Data Min Knowl Discov, 2011, 23: 447-478.
- [6] CHEN M, YANG Q, TANG X. Directed graph embedding[C] // Proceedings of the 20th International Joint Conference on Artificial Intelligence. Hyderabad, 2007: 2707-2712.
- [7] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[C] // Proceedings of Advances in Neural Information Processing Systems. Lake Tahoe, 2013: 3111-3119.
- [8] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space[J]. arXiv: 1301. 3781.
- [9] MIKOLOV T, KARAFIAT M, BURGET L, et al. Recurrent neural network based language model[C] // Proceedings of International Speech Communication Association. 2010: 1045-1048.
- [10] PEROZZI B, AL-RFOU R, SKIENA S. Deepwalk: online learning of social representations[C] // The ACM SIGKDD International Conference. ACM, 2014: 701-710.
- [11] GROVER A, LESKOVEC J. node2vec: scalable feature learning for networks[C] // The ACM SIGKDD International Conference. 2016.
- [12] CHENG W, GREAVES C, WARREN M. From n-gram to skip-gram to concgram[J]. International Journal of Corpus Linguistics, 2006, 11(4): 411-433.
- [13] TANG J, QU M, WANG M, et al. Line: large-scale information network embedding[C] // International Conference on World Wide Web. 2015: 1067-1077.
- [14] CAO S, LU W, XU Q. Grarep: Learning graph representations with global structural information[C] // KDD. 2015.
- [15] ZHANG Z, CUI P, WANG X, et al. Arbitrary-order proximity preserved network embedding[C] // Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 2778-2786.
- [16] OU M, CUI P, PEI J, et al. Asymmetric transitivity preserving graph embedding[C] // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, 2016, 1: 1105-1114.
- [17] MA J, CUI P, WANG X, et al. Hierarchical taxonomy aware network embedding[C] // Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018: 1920-1929.
- [18] WANG X, CUI P, WANG J, et al. Community Preserving Network Embedding[C] // AAAI. 2017.
- [19] CAVALLARI S, ZHENG V W, CAI H, et al. Learning community embedding with community detection and node embedding on graphs[C] // Proceedings of CIKM. 2017.
- [20] TU C C, ZENG X K, WANG H, et al. A Unified Framework for Community Detection and Network Representation Learning[J]. arXiv: 1611. 06645.
- [21] LE T M, LAUW H W. Probabilistic latent document network embedding[C] // 2014 IEEE International Conference on Data Mining (ICDM). IEEE, 2014: 270-279.
- [22] CHANG J, BLEI D M. Relational topic models for document networks[C] // International Conference on Artificial Intelligence and Statistics. 2009: 81-88.
- [23] YANG C, LIU Z, ZHAO D, et al. Network representation learning with rich text information[C] // International Conference on Artificial Intelligence. 2015: 2111-2117.
- [24] NATARAJAN N, DHILLON I S. Inductive matrix completion for predicting gene-disease associations[J]. Bioinformatics, 2014, 30(12): i60-i68.
- [25] AHMED N K, ROSSI R A, ZHOU R, et al. Inductive Representation Learning in Large Attributed Graphs[J]. arXiv: 1710. 09471v2, 2017.
- [26] AHMED N K, ROSSI R A, ZHOU R, et al. A Framework for Generalizing Graph-based Representation Learning Methods[J]. arXiv: 1709. 04596, 2017.
- [27] NGUYEN D, MALLIAROS F D. BiasedWalk: Biased Sampling for Representation Learning on Graphs[J]. arXiv: 1809. 02482, 2018.
- [28] TU C C, LIU H, LIU Z Y, et al. CANE: context-aware network embedding for relation modeling[C] // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics.

- tics, Vancouver, 2017; 1722-1731.
- [29] PAN S, WU J, ZHU X, et al. Tri-Party Deep Network Representation[C]// The 25th International Joint Conference on Artificial Intelligence (IJCAI-2016). AAAI Press, 2016.
- [30] ZHU J, AHMED A, XING E P. Medlda; maximum margin supervised topic models[J]. JMLR, 2012, 13(1): 2237-2278.
- [31] TU C C, ZHANG W C, LIU Z Y, et al. Max-Margin DeepWalk: discriminative learning of network representation[C]// Proceedings of International Joint Conference on Artificial Intelligence (IJCAI). 2016.
- [32] HEARST M A, DUMAIS S T, OSMAN E, et al. Support vector machines[J]. IEEE Intelligent Systems and their Applications, 1998, 13(4): 18-28.
- [33] HUANG X, LI J, HU X. Label Informed Attributed Network Embedding[C]// Tenth Acm International Conference on Web Search & Data Mining. ACM, 2017.
- [34] 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.
- [35] LIU J, HE Z C, WEI L, et al. Content to Node: Self-translation Network Embedding[C]// SIGKDD. 2018.
- [36] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.
- [37] WANG D X, CUI P, ZHU W W. Structural deep network embedding[C]// Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2016; 1225-1234.
- [38] ZHANG Z, YANG H X, BU J J, et al. ANRL: Attributed Network Representation Learning via Deep Neural Networks [C]// IJCAI. 2018.
- [39] KIPF T N, WELING M. Semi-supervised classification with graph convolutional networks[J]. arXiv; 1609. 02907, 2016.
- [40] KIPF T N, WELING M. Variational graph auto-encoders[J]. arXiv; 1611. 07308, 2016.
- [41] KINGMA D P, WELING M. Auto-encoding variational bayes [C]// Proceedings of the International Conference on Learning Representations (ICLR). 2014.
- [42] WANG S, TANG J, AGGARWAL C, et al. Signed network embedding in social media[C]// Proceedings of the 2017 SIAM International Conference on Data Mining. SIAM, 2017; 327-335.
- [43] TU K, CUI P, WANG X, et al. Deep recursive network embedding with regular equivalence[C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018; 2357-2366.
- [44] ZHU D, CUI P, WANG D, et al. Deep variational network embedding in wasserstein space[C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2018; 2827-2836.
- [45] VELIKOVI P, CUCURULL G, CASANOVA A, et al. Graph attention networks[J]. arXiv; 1710. 10903, 2017.
- [46] HAMILTON W, YING Z, LESKOVEC J. Inductive representation learning on large graphs[C]// Advances in Neural Information Processing Systems. 2017; 1024-1034.
- [47] KULKARNI V, AL-RFOU R, PEROZZI B, et al. Statistically Significant Detection of Linguistic Change[C]// Proceedings of the 24th International Conference on World Wide Web. ACM, 2014; 625-635.
- [48] HAMILTON W L, LESKOVEC J, JURAFSKY D, et al. Diachronic word embeddings reveal statistical laws of semantic change[J]. Meeting of the Association for Computational Linguistics, 2016, 1; 1489-1501.
- [49] LI J, DANI H, HU X, et al. Attributed Network Embedding for Learning in a Dynamic Environment[C]// Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, 2017; 387-396.
- [50] STEWART G W. Matrix Perturbation Theory [M] // Matrix Perturbation Theory. 1990.
- [51] HARDOON D R, SZEDMAK S, SHAW-TAYLOR J. Canonical correlation analysis: An overview with application to learning methods[J]. Neural computation, 2004, 16(12): 2639-2664.
- [52] GOYAL P, KAMRA N, HE X, et al. Dyngem: Deep embedding method for dynamic graphs[J]. arXiv; 1805. 11273, 2018.
- [53] CHANG S, HAN W, TANG J, et al. Heterogeneous network embedding via deep architectures[C]// Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015; 119-128.
- [54] NGUYEN G H, LEE J B, ROSSI R A, et al. Continuous-time dynamic network embeddings[C]// Companion of the The Web Conference 2018 on The Web Conference 2018. 2018; 969-976.
- [55] ZHOU L, YANG Y, REN X, et al. Dynamic Network Embedding by Modeling Triadic Closure Process[C]// AAAI. 2018.
- [56] LUN D, YUN W, GUOJIE S, et al. Dynamic Network Embedding: An Extended Approach for Skip-gram based Network Embedding[C]// IJCAI. 2018.
- [57] ZHU D, CUI P, ZHANG Z, et al. High-order Proximity Preserved Embedding For Dynamic Networks[J]. IEEE Transactions on Knowledge & Data Engineering, 2018, 30(11): 2132-2144.



**DING Yu**, born in 1989, doctoral student. His main research interests include artificial intelligence and network security.



**WEI Hao**, born in 1990, Ph. D. His main research interests include complex network in machine learning, network embedding, online time series prediction.