基于元路径的动态异质网络表示学习

刘 群,谭洪胜,张优敏,王国胤
(重庆邮电大学计算机科学与技术学院,重庆 400065)

摘 要: 对网络表示学习的研究已经取得了很多成果,但是大部分网络表示学习模型忽略了网络动态性和异质性,无法区分网络中耦合的时间和空间(结构)特征,也不能捕获网络的丰富语义信息. 本文提出了基于元路径的动态异质网络表示学习方法. 首先将节点的邻域结构按照时间划分出不同的子空间结构,并为每个节点采样出所有时间加权元路径的序列. 其次通过门控循环单元将节点的全部时间加权元路径序列上的邻域信息进行集成,最后利用带注意力机制的双向门控循环单元对融合后的节点序列进行时空上下文信息学习,获得每个节点的最终表示向量. 通过在真实数据集上的实验表明,在节点分类、聚类和可视化的下游任务测试中,本文提出的算法较基线方法在性能上均有较大提升. 节点分类任务中的 Micro-F1 平均提高了 1.09%~3.72%,节点聚类任务中的 ARI 值提高了 3.23% ~ 14.49%.

关键词: 网络表示学习; 动态异质网络; 元路径; 注意力机制; 门控循环单元
中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112(2022)08-1830-10
电子学报 URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20211288

Dynamic Heterogeneous Network Representation Method Based on Meta-Path

LIU Qun, TAN Hong-sheng, ZHANG You-min, WANG Guo-yin
(*College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing* 400065, *China*)

Abstract: The researches of network representation learning have made many achievements. Since most of the researches ignore the dynamics and heterogeneity of the networks, coupled temporal and spatial structure features can not be distinguished, and rich semantic information of the network cannot be captured well. In this paper, meta-path based dynamic heterogeneous network representation learning method is proposed. Firstly, the neighborhood structures of nodes are divided into different sub-spaces according to their time, then the sequences of all time-weighted meta-paths for each node are sampled. Secondly, the neighborhood information on all time-weighted meta-paths of each node is integrated by a gated recurrent unit network(GRU). Furthermore, a bi-directional gated recurrent unit network(Bi-GRU) with an attention mechanism is used to learn the spatio-temporal contextual information from the merged sequences, and the final node representation will be received. Experiments on real data sets show that our algorithm has greatly improved performance on the downstream network tasks, such as node classification, clustering and visualization. Compared with state-of-the-art baseline methods,the Micro-F1 value has been raised by 1.09%~3.72% averagely on classification tasks, and the ARI value has been increased by 3.23%~14.49% on clustering tasks.

Key words: network representation learning; dynamic heterogeneous network; metapath; attention mechanism; gated recurrent unit

1 引言

网络表示学习在保留网络结构和语义信息的同时,将网络中的节点表示成低维向量,以提升图挖掘后续任务的性能,如节点分类[1]、聚类[2]等. 现有大多数网络表示学习方法是为静态网络设计. 文献[3~5]通过特定的游走策略得到节点的节点序列,然后将其输入

收稿日期:2021-09-17;修回日期:2022-04-21;责任编辑:李勇锋
基金项目:国家自然科学基金重点项目(No.61936001);重庆市教委重点合作项目(No.HZ2021008);重庆市自然科学基金(No.cstc2021ycjh-bgzxm0013)

到自然语言模型Skip-gram[6]中,学习到节点的向量表示. LINE[7]保留了节点的一阶和二阶相似性进行概率建模,通过负采样算法提升了在大规模网络上学习节点表示的效率. 算法SDNE[8]采用深度学习的方法也取得了较好的效果. 随着图神经网络[9](Graph Neural Network, GNN)的广泛应用,很多网络表示学习方法[10~13]聚合邻居节点信息和节点本身的属性信息,以提升节点向量表示质量. 但是上述方法忽略了网络空间信息(拓扑结构和属性)随时间变化的特点,而只是简单地将不同时间对应的空间压缩在一起.

由于网络是随时间不断变化的,前一秒没有关系的两个节点可能会在下一秒关联,而节点之间边的建立也改变了网络的拓扑结构. 因此,只考虑静态的处理方式不符合网络实际的演化规律. 早期的动态同质网络表示学习结合网络演化特点,通过快照的方式获得网络在不同时间上的动态演化信息[14,15]. 不同于上述基于快照的模型,CTDNE[16]设计了能捕获网络时间信息的游走序列,可以更细粒度地捕捉网络中重要信息. M^{2}DNE[17]从微观动力学和宏观动力学两个角度很好地模拟了网络演化过程. 动态同质网络表示学习未考虑网络节点和边的差异,如果将其直接应用于动态异质网络中,将不可避免地丢失语义信息. 目前,典型的动态异质网络表示学习方法[18~20],使用快照对空间(结构)进行划分,前提是要确保空间子图之间的平滑演化(节点和边微小变化). 但是,在真实网络中(例如学术网络),子图之间的节点和边存在巨大差异,基于快照的划分方式会将交互的时间戳删除,不仅导致网络的形成过程变得未知,还将导致空间子图之间的相关性变低.

为了解决现有方法的不足,本文提出了一种基于元路径的动态异质网络表示学习方法DHNR,主要贡献有以下几点.(1)为捕获更精细的语义,提出了时间加权元路径对耦合的时空结构进行划分. 通过对时间进行编码,把每一条元路径对应的时间信息编码到元路径序列里.(2)不同于大部分元路径处理方法,DHNR通过设计门控循环单元(Gated Recurrent Unit Network, GRU)将每条元路径序列中的所有邻居节点属性和信息都聚合到节点序列中,而不是仅仅保留两个末端节点的结构信息,从而保证尽量完整地捕获节点的上下文语义.(3)拓展了循环神经网络(Recurrent Neural Network, RNN),设计了带注意力机制的双向门控循环单元(Bi-directional Gated Recurrent Unit Network, Bi-GRU),对不同时间的元路径序列按照重要性进行权重分配,进而完成聚合.(4)三个真实网络数据集上的实验证明了本文模型优于其他基线模型. 在节点分类任务中,Micro-F1平均提高了1.09%~3.72%,节点聚类任务的ARI值提高了3.23%~14.49%. 同时在对网络动态特性捕获和消融分析实验上,也证明了本文模型的有效性.

2　相关定义

定义1　动态异质网络　动态异质网络可以形式化表示为$G=(V, E, T)$,其中V表示节点集合,E表示连边集合,T表示边上时间集合. 网络中节点和节点类型间的映射函数为$\varphi: V\to\mathcal{A}$,边和边类型间的映射函数为$\psi: E\to\mathcal{R}$ 其中$\mathcal{A}$和$\mathcal{R}$表示节点类型和边类型,动态异质网络中$|\mathcal{A}|+|\mathcal{R}|\geqslant 2$. 每条边$(i, j, t)\in E$表示$t$时刻节点$i$连接到节点$j$.

定义2　元路径[21]　一条元路径Φ可以表示为$A_1\xrightarrow{r_1}A_2\xrightarrow{r_2}\cdots\xrightarrow{r_m}A_{m+1}$(缩写为$A_1A_2\cdots A_{m+1}$). $r=r_1\circ r_2\circ\cdots\circ r_m$定义了$A_i$和$A_{m+1}$之间的复合关系,$\circ$表示$A_i$和$A_j$之间关系的复合操作符.

图1给出了学术网络的示意图. 该网络包含t_1,t_2两个时间点对应的空间. 在t_1时,A_1、A_2、A_3、A_4四位作者在C_1,C_2两个会议上发表了三篇论文P_1、P_3、P_4;在t_2时,A_1、A_2和A_3三位作者在会议C_1上再次合作发表了论文P_2. 传统的基于元路径的方法由于忽略网络中的时间因素,容易导致一条元路径序列上的节点间存在空间(结构)耦合. 例如,在元路径APCPA引导下,捕获到$A_1\xrightarrow{t_1}P_1\xrightarrow{t_1}C_1\xrightarrow{t_2}P_2\xrightarrow{t_2}A_2$这样跨时间的节点序列. 该序列耦合了两个时间点下不同的结构空间,无法学习到序列里面的时间信息,同时也无法保证语义的准确性. 根据网络平滑演变假设[7]可知,间隔越近,其对应的空间信息也会越相似,其中的微妙变化只有对元路径考虑时间才能精细捕获. 为了解决此问题,本文提出了时间加权元路径.

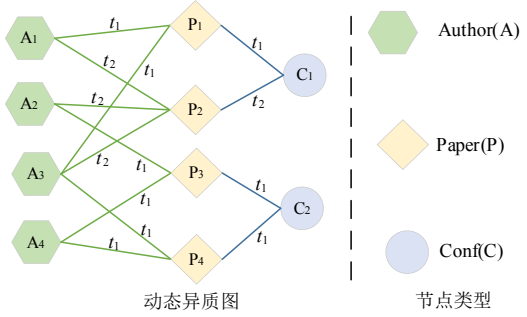


图1　动态异质网络示例

定义3　加权元路径[22]　加权元路径s是基于某一属性值约束关系的元路径,记为$A_1\xrightarrow{\delta_1(r_1)}A_2\xrightarrow{\delta_2(r_2)}\cdots\xrightarrow{\delta_m(r_m)}A_{m+1}|C$(也可记为$A_1\left(\delta_1\left(r_1\right)\right)A_2\left(\delta_2\left(r_2\right)\right)\cdots A_m\left(\delta_m\left(r_m\right)\right)A_{m+1}|C$). $\delta(r)$是属

性值函数，表示定义在关系 r 上的一个属性值. $A_i \xrightarrow{\delta_i(r_i)} A_{i+1}$ 表示 A_i 与 A_{i+1} 的边关系为 r_i，其属性值为 $\delta_i(r_i)$. C 表示属性值函数之间的相关性约束. 如果元路径中的所有属性值函数均为空集(相应的约束 C 也是空集)，则该路径为未加权元路径，否则该路径称为加权元路径. 以图1为例，作者A与论文P的时间属性取值为 t_1 与 t_2. 加权元路径 $A \xrightarrow{t_1} P$ 表示作者在 t_1 时发表论文. 若在 t_1 时间，两位作者 A_i 和 A_j 在会议 C_j 上发表论文 P_i 和 P_j，该加权元路径为 $A_i \xrightarrow{a} P_i \xrightarrow{a} C_j \xrightarrow{b} P_j \xrightarrow{b} A_j | a=\mathrm{b}=t_1$.

定义4 时间加权元路径 为了捕获动态异质网络上的时间信息，并对耦合的空间进行划分，本文将每一条加权元路径约束为同一时间值 t 下的加权元路径，称作时间加权元路径 s_t，可形式化表示为 $s_t = A_1 \xrightarrow{\delta_1(r_1)} A_2 \xrightarrow{\delta_2(r_2)} \cdots \xrightarrow{\delta_m(r_m)} A_{m+1} | \left(\delta_1(r_1)=\delta_2(r_2)=\cdots=\delta_m(r_m)=t\right)$，其中 $t \in T$.

通过不同的时间属性值，可以得到节点在不同时间下的加权元路径序列 H_t^i，H_t^i 表示第 i 种类型元路径下时间为 t 的元路径序列，每个序列保留了节点语义信息和结构信息. 若 A_1 是该序列中的根节点，第 i 层节点是 A_i，则它也是根节点 A_1 的第 i−1 跳邻居.

3 基于元路径的动态异质网络表示

本文提出的基于元路径的动态异质网络表示学习模型框架如图2所示，该模型由三部分构成：空间划分，邻域信息聚合，时序信息集成. 空间划分模块根据时间将每个节点的结构邻域划分为不同时间下的时间加权元路径序列. 通过GRU模型和相对时间编码，邻域信息融合模块能够尽量保留网络的完整语义和时间信息. 最后时序信息集成模块使用带注意力机制的Bi-GRU模型进行过滤筛选.

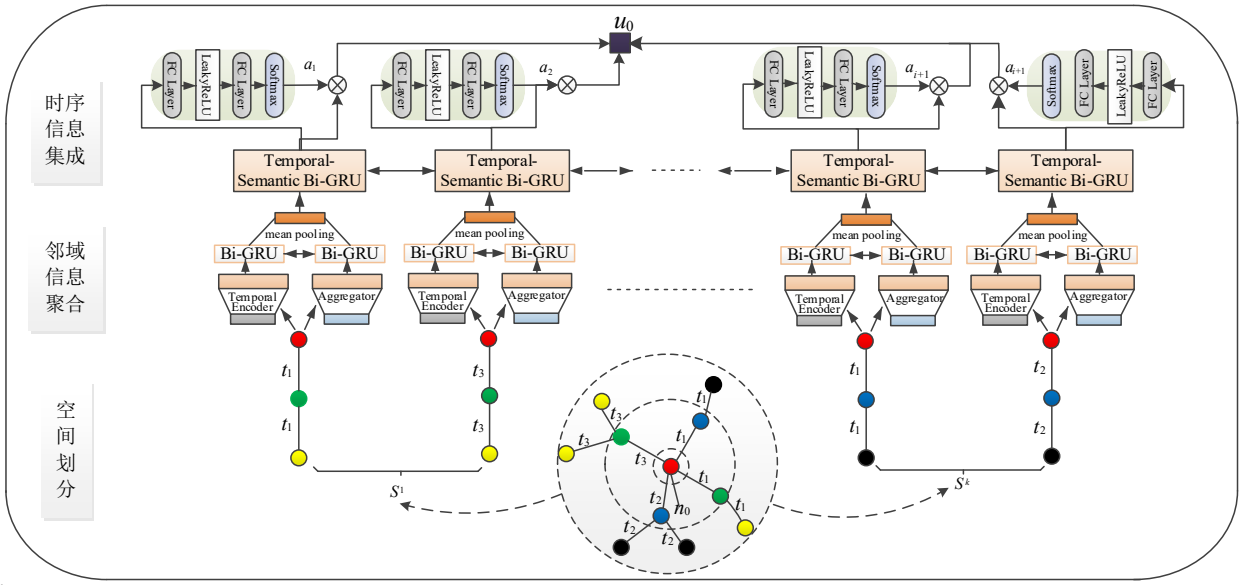


图2 模型框架示意图

3.1 空间划分

DHNR采用时间加权元路径为节点采样邻居序列. 图3说明了将网络通过时间加权元路径划分后，获得不同时间值下多条时间加权元路径序列的过程. 例如，给定一个节点类型为 $A_1=\varphi(n_0)$ 的节点，在时间加权元路径 s_{t_0} 的引导下，首先采样在 t_1 时与节点 n_0 建立 r_1 类型边的邻居节点 n_1，然后采样在 t_1 时与节点 n_1 建立 r_2 类型边的邻居节点 n_2，如此重复，直至采样与 n_m 在时间 t_1 建立 r_m 类型边的邻居节点 n_{m+1}. 最后获得节点 n_0 在时间属性值为 t_1 的一条时间加权元路径序列. 通过设置不同的时间加权元路径的时间值，可以将网络耦合的空间进行有效划分.

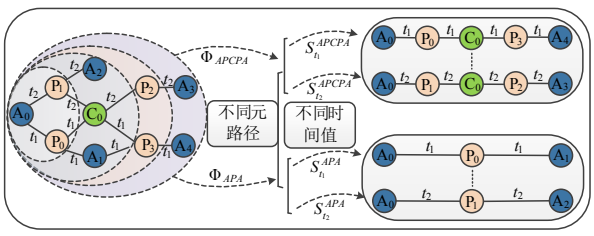


图3 空间划分过程

为了进一步提取动态异质网络中的丰富语义信息，DHNR选取k条不同类型的元路径$\Phi_1, \Phi_2, \cdots, \Phi_k$，为每种类型的元路径构造数量为$w$的不同时间加权元路径，$S_t^k = \{\{s_{t_1}^1, s_{t_2}^1, \cdots, s_{t_{w_1}}^1\}, \cdots, \{s_{t_1}^k, s_{t_2}^k, \cdots, s_{t_{w_k}}^k\}\}$，其中$\forall s_j^i \in S_t^k$，$t_i$表示时间值，$\forall t_i \in T$. s_t^j表示为第j种时间属性值为t的时间加权元路径. 最终获取到节点的时间加权元路径序列集合$\left\{\{H_1^1, H_2^1, \cdots, H_{W_1}^1\}, \cdots, \{H_1^k, H_2^k, \cdots, H_{W_k}^k\}\right\}$.

3.2　邻域信息聚合

为了捕获节点的语义信息，DHNR将每个节点的不同时间加权元路径信息分别进行聚合，如图4所示. 以节点n_0为例，给定一条由时间加权元路径s_t^j提取的n_0的时间加权元路径序列H_t^j，以及该序列上每个节点的初始特征向量$\boldsymbol{x}_i \in \boldsymbol{R}^d$($d$表示初始特征向量维数). 将$n_0$的时间加权元路径序列中所有邻居信息及其自身信息聚合起来，形成一个初始的向量表示.

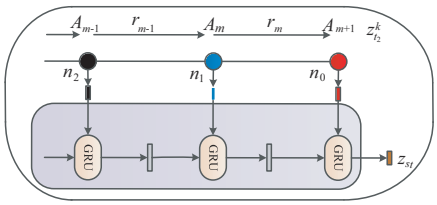


图4　邻域信息聚合

传播过程中的邻居信息可以视为序列输入，而GNN不能处理序列数据，所以DHNR模型引入了GRU构造的传播模块，将节点n_0本身特征x_0及其邻域信息编码成特定时间特定语义下的向量$\boldsymbol{z}_0^{st}$，st表示序列语义为s，时间值为t的序列.

由于节点的异质性，不同类型的节点具有不同的特征空间. 对每种类型的节点设计特定类型的变换矩阵(例如节点类型为ϕ_i的变换矩阵为$\boldsymbol{M}_{\phi_i}$)，以此将不同类型节点的特征投影到相同的特征空间中. 转换过程如下：

$$\hat{\boldsymbol{x}}_i = \boldsymbol{M}_{\phi_i} \cdot \boldsymbol{x}_i \tag{1}$$

其中$\boldsymbol{x}_i$和$\hat{\boldsymbol{x}}_i$分别是节点n_i的原始特征和投影特征. 通过式(1)的投影操作，GRU传播模块可以处理序列中任意类型的节点. 传播模块在时间加权元路径序列上的基本递归过程如下：

$$\boldsymbol{h}_i^{(1)} = \hat{\boldsymbol{x}}_i, \quad \mathcal{A}_1 = \phi(n_i) \tag{2}$$

$$\boldsymbol{h}_i^{(a)} = \mathrm{GRU}(\hat{\boldsymbol{x}}_i, \boldsymbol{h}_i^{a-1}), \quad \mathcal{A}_a = \phi(n_i) \tag{3}$$

其中$0 < a \leqslant m+1$，$\boldsymbol{h}_i^{(a)}$表示节点n_i通过GRU输出的隐藏状态，节点n_i是节点n_0在时间加权元路径序列上的$m+1-a$跳邻居节点. $\mathrm{GRU}(\hat{\boldsymbol{x}}_i, \boldsymbol{h}_i^{a-1})$公式表示为：

$$\boldsymbol{z}_i = \sigma(\boldsymbol{A}_z \hat{\boldsymbol{x}}_i + \boldsymbol{B}_z \boldsymbol{h}_i^{a-1}) \tag{4}$$

$$\boldsymbol{r}_i = \sigma(\boldsymbol{A}_r \hat{\boldsymbol{x}}_i + \boldsymbol{B}_r \boldsymbol{h}_i^{a-1}) \tag{5}$$

$$\hat{\boldsymbol{h}}_i^a = \tanh(\boldsymbol{A}_h \hat{\boldsymbol{x}}_i + \boldsymbol{B}_h (\boldsymbol{r}_i \circ \boldsymbol{h}_i^{a-1})) \tag{6}$$

$$\boldsymbol{h}_i^{(a)} = \boldsymbol{z}_i \circ \boldsymbol{h}_i^{a-1} + (1 - \boldsymbol{z}_i) \circ \hat{\boldsymbol{h}}_i^a \tag{7}$$

其中，$\boldsymbol{A}_i \in \boldsymbol{R}^{d \times d'}$和$\boldsymbol{B}_j \in \boldsymbol{R}^{d \times d'}$是参数，$\boldsymbol{z}_i, \boldsymbol{r}_i$是更新门和遗忘门. $\circ$表示逐元素乘法. 在时间加权元路径序列的结构邻域上传播m+1次后，节点n_0在时间加权元路径s_t^j上的状态向量输出可由下面公式获得：

$$\boldsymbol{z}_0^{st} = \mathrm{GRU}(\hat{\boldsymbol{x}}_0, \boldsymbol{h}_i^m) \tag{8}$$

其中$\boldsymbol{z}_0^{st} \in \boldsymbol{R}^{d'}$($d'$是特征维度). 每种类型的时间加权元路径序列，都设计了特定类型的GRU聚合模块(同类时间加权元路径序列共享GRU参数). 由于每个序列都对应一个独立的时间，因此还需要对每个序列上的时间进行编码. 时间编码器要具有泛化到不可看见时间的能力，DHNR采用了相对时间编码技术[23]作为时间编码器，定义了一组固定的正弦函数作为时间偏置，如下所示：

$$\mathrm{Base}(t, 2i) = \sin(t/10000^{\frac{2i}{d}}) \tag{9}$$

$$\mathrm{Base}(t, 2i+1) = \cos(t/10000^{\frac{2i+1}{d}}) \tag{10}$$

$$\boldsymbol{z}_0^{RT(t)} = \mathrm{T\_Linear}(\mathrm{Base}(t)) \tag{11}$$

其中t为时间加权元路径序列的时间属性值，$\boldsymbol{z}_0^{RT(t)} \in \boldsymbol{R}^{d'}$为经过T-Linear线性微调变换后节点的时间特征向量. 最后，将时间特征向量$\boldsymbol{z}_0^{RT(t)}$与结构语义特征向量$\boldsymbol{z}_0^{st}$进行聚合. 不同于传统直接拼接的思想，考虑到Bi-GRU及其变体[17]结构简单，可以融合异质属性信息和表达能力强的特点，DHNR采用参数量小的Bi-GRU来获取节点的深层次交互特征. 计算方法如下：

$$\boldsymbol{z}_0^{s_t^j} = \mathrm{Mean}(\overleftrightarrow{\mathrm{Bi\text{-}GRU}}_{\boldsymbol{z}_0^i \in \{\boldsymbol{z}_0^{st}, \boldsymbol{z}_0^{RT(t)}\}}\{\boldsymbol{z}_0^i\}) \tag{12}$$

其中$\boldsymbol{z}_0^i$是节点的时间特征和语义结构特征. 将其作为Bi-GRU的输入，然后通过Mean函数对节点n_0所有状态特征进行平均，从而得到节点在一条时间加权元路径序列上包含有时间编码的向量表示$\boldsymbol{z}_0^{s_t^j} \in \boldsymbol{R}^{d'}$.

3.3　时序信息集成

为了捕获更多的语义信息，同时学习的网络演化规律，DHNR将所有序列的特征向量进行融合，形成节点的最终表示. DHNR拓展了已有的RNN模型，运用带注意力机制的Bi-GRU进行深层次信息交互，为不同时间元路径序列下的节点表示向量分配不同权重，再聚合得到节点的最终表示向量.

由于邻域信息聚合的是每个时间加权元路径上的邻域节点信息，考虑到节点的演化规律需要结合节点在不同时空结构下的语义特征信息，DHNR使用Bi-GRU来交互不同时间下的特征信息，以此来模拟网络的演化. 其公式为：

$$\{\boldsymbol{o}_1,\boldsymbol{o}_2,\cdots,\boldsymbol{o}_g\}=\overleftrightarrow{\text{Bi-GRU}}_{(z_0^i\in \boldsymbol{Z}_0)}\{\boldsymbol{z}_0^i\} \tag{13}$$

其中$\{\boldsymbol{o}_1,\boldsymbol{o}_2,\cdots,\boldsymbol{o}_g\}$是Bi-GRU的状态向量序列集合，$g$为节点采样的序列数. 传统方法直接拼接这些状态向量作为节点的最终向量表示，这种处理方式无法关注到重要特征. 本文模型引用注意力机制对特征向量进行权重设计，注意力权重α_i的计算公式如下：

$$\boldsymbol{\alpha}_i=\frac{\exp\{\text{LeakyReLU}(\boldsymbol{a}^{\mathrm{T}}\boldsymbol{o}_i)\}}{\sum_{i=1}^{g}\{\exp\{\text{LeakyReLU}(\boldsymbol{a}^{\mathrm{T}}\boldsymbol{o}_i)\}\}} \tag{14}$$

$\boldsymbol{\alpha}_i$越高，$\boldsymbol{o}_i$则越重要. LeakyReLU为激活函数，$\boldsymbol{a}\in\boldsymbol{R}^{1\times 2d'}$为注意力参数，则节点$n_0$的所有状态特征聚合得到的节点最终向量表示为：

$$\boldsymbol{u}_0=\sum_{\boldsymbol{o}_i\in\boldsymbol{O}}\boldsymbol{\alpha}_i\cdot\boldsymbol{o}_i \tag{15}$$

$\boldsymbol{u}_0\in\boldsymbol{R}^{d'}$是节点$n_0$的最终向量表示. 在节点最终表示向量的生成过程中，采用了最小化所有标记节点的真实值和预测值的交叉熵作为损失函数：

$$\mathcal{L}=-\sum_{l\in Y_l}y_l\ln\hat{y}_l \tag{16}$$

其中Y_l是节点真实标签集合，y_l和$\hat{y}_l$表示节点标签真实值和节点标签预测值. DHNR算法如下.

算法1 基于元路径的动态异质网络表示学习

输入：动态异质网络$G=(V,E,T)$，节点特征$\{x_i,\forall i\in V\}$，时间加权元路径集合$\{\{s_{t_1}^1,s_{t_2}^1,\cdots,s_{t_{w_1}}^1\},\cdots,\{s_{t_1}^k,s_{t_2}^k,\cdots,s_{t_{w_k}}^k\}\}$

输出：节点最终表示$\boldsymbol{U}$

```
for epoch in len(epochs)
    for i ∈ V do
        for s_j^t ∈ {{s_{t_1}^1, s_{t_2}^1, ···, s_{t_{w_1}}^1}, ···, {s_{t_1}^k, s_{t_2}^k, ···, s_{t_{w_k}}^k}} do
            通过s_j^t查找节点时间加权元路径邻居H_J^t
        end
        获得{{H_1^1, H_2^1, ···, H_{W_1}^1}, ···, {H_1^k, H_2^k, ···, H_{W_k}^k}}
        for H_J^t ∈ {{H_1^1, H_2^1, ···, H_{W_1}^1}, ···, {H_1^k, H_2^k, ···, H_{W_k}^k}} do
            通过式(1)(2)(3)(8)计算每条序列表示s_i^{st}
            根据式(9)(10)(11)计算时间编码s_i^{RT(t)}
            根据式(12)得到每条时间加权元路径序列特征z_i^{s_i'}
        end
        获得节点在所有序列上的向量表示集合Z_0
        根据式(13)交互不同时间加权元路径序列特征信息
        根据式(14)计算状态向量的影响权重
        根据式(15)融合不同状态向量得到节点向量表示u_i
    end
end
return U
```

3.4 复杂度分析

DHNR模型为三个模块构成. 若网络中有N个节点，则三个模块分别对应的时间复杂度分析如下：空间划分模块的时间为Nl，其中每个节点采样时间为$l=w_1f_1+w_2f_2+\cdots\cdots+w_kf_k$，每条时间加权元路径序列节点数为$f_i(i=1,\cdots,k)$，$k$为元路径类型数，$w_i(i=1,\cdots,k)$表示每种类型元路径的数量，则每个节点采样的序列数为$g=w_1+w_2+\cdots\cdots+w_k$；邻域信息聚合模块的时间为$N(7ld^2+3ld+3gd^2)$，其中$d$为向量维度；时序信息集成模块的时间为$Ng(12d^2+10d+1)$. 由于$d$、$g$、$l$都为很小的常数，最终DHNR的时间复杂度为$O(N)$.

4 实验分析

为了验证DHNR模型的性能，针对三种典型的网络挖掘任务，与不同基线模型进行了对比，同时也对DHNR模型捕获网络动态性的能力做了验证测试.

4.1 数据集

AMiner[①]和DBLP[②]是广泛用于异质网络研究的两个数据集. DBLP-4是从DBLP中抽取生成的一个公开数据集. 为了更充分地进行对比，本文对DBLP数据集抽取生成了数据子集DBLP-10. 表1归纳了三个数据集的统计特征.

表1 数据集描述

数据集	节点类型	节点数	时间	标签	元路径
AMiner	Paper(P)	18 181	16	5	APA APCPA
	Conference(C)	22			
	Author(A)	22 942			
DBLP-4	Paper(P)	14 328	4	4	APA APCPA APTPA
	Conference(C)	20			
	Author(A)	4 057			
	Term(T)	8 898			
DBLP-10	Paper(P)	8 970	6	10	PAP PCP
	Conference(C)	51			
	Author(A)	15 019			

(1)AMiner：是一个学术网络数据集，实验抽取了从1990年到2005年且作者在其中五个领域研究方向有变化的子集. 对于每位在这五个领域发表文章的作者，其标签与其主要研究领域相符.

(2)DBLP-4：是一个计算机科学的学术网络，包含四类节点，根据作者的研究方向将其分为四个领域.

(3)DBLP-10：该数据集包含从2008年到2013年10个研究领域发表的文章，标签是根据文章被发表的会议或者期刊所属领域进行标记.

4.2 对比算法

将DHNR算法与以下基线算法进行对比：

(1)DeepWalk[4]和GCN[13]：这是两种静态同质网络

①https://www.aminer.cn/data

②https://dblp.uni-trier.de

表示学习模型. 前者基于随机游走生成序列,再学习节点表示,简写为DeW. 后者为图卷积神经网络,在实验对比中,测试了本文使用的所有元路径,记录其中的最佳结果.

(2)Metapath2vec[3]和HAN[10]:这是两种静态异质网络的表示学习方法,前者通过元路径约束生成的序列来学习节点表示,简写为M2v. 后者考虑了节点和语义级两种注意力. 本文实验中,HAN元路径的选择和本文相同.

(3)M^2DNE[17]:一种动态同质网络表示学习方法,用微观动力学和宏观动力学模拟网络的演化.

(4)DHNE[18]和DyHATR[20]:都是基于快照的动态异质网络表示模型. DHNE在历史快照和当前快照进行元路径约束游走,通过改进的Skip-gram模型学习动态异质网络的节点表示. DyHATR利用层次注意力来学习网络的异质性,并利用带注意力机制的RNN学习网络演化过程.

(5)$DHNR_{stu}$:本文模型DHNR的变体,通过消除每条序列上的时间编码,进行消融实验对比.

实验过程中使用Par2vec[24]对节点的文本内容进行预训练,设置初始特征维数为128. 在DHNR模型中,节点隐藏层维度和最终表示维度都设置为128维. 使用Adam优化器对参数进行优化,学习率设置为0.000 1. 为了对比公平,其他基线模型的节点表示维度均设置为128维,Deepwalk和Metapath2vec设置其窗口大小参数5,其他超参数设置为各自论文中指定值. 所有实验都重复10次,取平均值作为最终实验结果. 对比的静态网络表示学习模型,实验中忽略了网络中的时间属性值. 对比的动态同质表示学习模型,将所有节点和边视为同一类. 对比的动态异质网络表示学习算法,依据其思想,构造了不同的时间快照进行学习.

4.3　节点分类

实验中,在AMiner和DBLP-4数据集上是对作者节点类型进行分类. 在DBLP-10数据集上,是对论文节点进行分类. 采用的分类算法是逻辑回归分类模型. 将DHNR学习到的节点向量表示作为逻辑回归分类器的输入. 实验过程中训练集的比例设置为20%,50%,80%.

表2给出了以Micro-F1和Macro-F1作为评价指标的分类结果. 从实验结果可以看出,DHNR在三个数据集上的表现优于所有的对比算法,证明了DHNR学习的节点向量表示的有效性. 在DBLP-4数据集上,由于该数据集包含了关键词等类型节点,使网络中节点类型更加丰富,节点相互之间产生的边也更多,语义信息也更丰富. 因此要想获得更多更准确的语义信息,需要设计更多的元路径. 实验对比中,发现M^2DNE表现性能好于本文模型,分析原因其核心思想是计算邻居节点间的相似性,较好地适用于边丰富的网络. 但是当训练比例稍大一点,本文模型DHNR表现更优. 这说明了DHNR不仅能获得较丰富语义信息,还能获得高阶邻居结构信息. 对比不同的动态异质网络表示学习,DHNR也展现最佳性能,说明DHNR通过元路径保留网络语义和结构信息同时,拓展的RNN很好地归纳聚合了网

表2　节点多分类结果

数据集	分类指标	训练比例	DeW	GCN	M2v	HAN	M^2DNE	DHNE	DyHATR	$DHNR_{stu}$	DHNR
AMiner	Macro-F1	20%	0.969 8	0.972 7	0.974 1	0.960 5	0.965 2	0.845 3	0.971 0	0.972 7	**0.976 2**
		50%	0.971 1	0.974 4	0.974 3	0.962 1	0.965 7	0.844 9	0.972 1	0.974 1	**0.976 8**
		80%	0.970 1	0.974 7	0.974 8	0.964 4	0.967 3	0.846 1	0.972 8	0.973 4	**0.977 1**
	Micro-F1	20%	0.972 1	0.974 9	0.976 5	0.964 2	0.967 3	0.853 9	0.972 7	0.975 2	**0.977 5**
		50%	0.973 4	0.975 8	0.976 7	0.966 3	0.967 9	0.855 6	0.973 5	0.975 8	**0.978 8**
		80%	0.972 5	0.976 6	0.977 4	0.968 1	0.969 1	0.855 5	0.974 9	0.975 6	**0.979 3**
DBLP-4	Macro-F1	20%	0.876 9	0.909 3	0.923 2	0.925 1	**0.940 1**	0.915 7	0.922 7	0.934 1	0.939 1
		50%	0.907 4	0.918 1	0.922 9	0.926 2	0.941 3	0.891 4	0.933 4	0.936 7	**0.943 1**
		80%	0.920 2	0.923 7	0.923 3	0.925 7	0.941 0	0.907 8	0.936 9	0.938 8	**0.943 9**
	Micro-F1	20%	0.885 8	0.915 1	0.928 5	0.930 1	**0.942 2**	0.924 5	0.926 1	0.937 7	0.941 2
		50%	0.915 4	0.914 4	0.927 9	0.931 4	0.943 4	0.910 3	0.935 2	0.941 5	**0.948 6**
		80%	0.926 2	0.919 7	0.928 0	0.941 0	0.944 6	0.926 2	0.944 3	0.943 6	**0.948 8**
DBLP-10	Macro-F1	20%	0.822 2	0.841 9	0.854 4	0.841 5	0.863 4	0.788 1	0.871 5	0.881 2	**0.897 2**
		50%	0.856 4	0.847 6	0.871 3	0.882 7	0.886 2	0.792 4	0.883 5	0.895 2	**0.898 4**
		80%	0.866 8	0.863 1	0.877 8	0.889 2	0.890 1	0.809 2	0.889 5	0.896 1	**0.899 3**
	Micro-F1	20%	0.838 7	0.874 3	0.862 1	0.885 1	0.881 8	0.777 1	0.872 4	**0.901 1**	0.897 4
		50%	0.852 1	0.892 1	0.871 9	0.899 4	0.887 1	0.784 1	0.891 0	0.903 8	**0.907 3**
		80%	0.856 6	0.896 4	0.880 5	0.901 2	0.892 3	0.821 8	0.899 2	0.909 4	**0.910 1**

络演化信息，提升了网络表示质量．同时为了验证本文模型DHNR不同模块的有效性，与不考虑时间编码的$DHNR_{stu}$进行了对比，可以看到，考虑了每个序列时间信息的DHNR，其分类性能有一定的提升．

4.4 节点聚类

实验中利用KMeans进行节点聚类，簇数K设置为数据集本身的标签数目．采用标准化互信息（NMI）和调整兰德系数（ARI）作为聚类评价指标．由于KMeans的性能受到初始质心的影响，进行了10次重复实验，最后实验结果取均值，表3给出了聚类的实验结果．

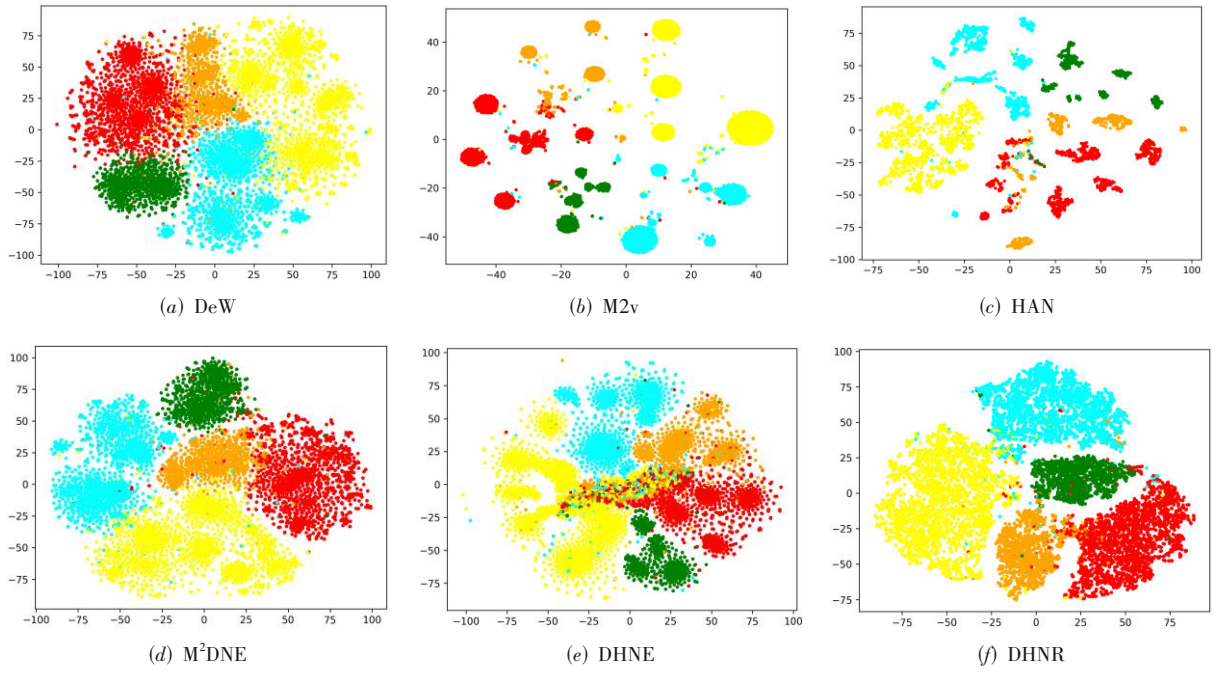
从表3可以看出，本文模型DHNR在聚类任务上具有较优的性能，比其他基线算法的兰德系数（ARI）高出3.23%～14.49%．DHNR区分了网络中的时间和空间，结合了节点在不同时间的轨迹，并在信息融合过程中考虑了路径上的中继节点，使得节点表示质量提升．

表3 节点聚类结果

对比算法	AMiner		DBLP-4		DBLP-10	
	NMI	ARI	NMI	ARI	NMI	ARI
DeW	0.782	0.691	0.761	0.811	0.404	0.212
GCN	0.879	0.907	0.724	0.770	0.396	0.483
M2v	0.727	0.591	0.780	0.830	0.294	0.171
HAN	0.787	0.810	0.781	0.835	**0.430**	0.540
M^2DNE	0.735	0.657	0.756	0.786	0.306	0.195
DHNE	0.174	0.456	0.238	0.093	0.152	0.061
DyHATR	0.823	0.870	0.768	0.822	0.376	0.419
$DHNR_{stu}$	0.911	**0.947**	0.784	0.843	0.382	0.551
DHNR	**0.916**	0.943	**0.810**	**0.862**	0.413	**0.553**

4.5 可视化

本节进一步将学习到的节点特征向量投影到二维空间中，进行可视化比较．采用t-SNE可视化了AMiner中的作者节点表示向量，根据作者节点的类型，进行不同着色．从图5可以看出，相比于其他基线模型，DHNR可以将不同类型的作者更好地映射到不同的社区，并且同一社区中节点聚集紧密，不同社区节点相距较远，只有极少数节点被嵌入到其他颜色区域，具有较高的聚类质量．



(a) DeW (b) M2v (c) HAN

(d) M^2DNE (e) DHNE (f) DHNR

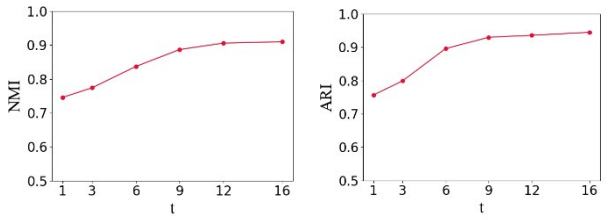
图5 AMiner网络中的节点向量表示的可视化

4.6 动态分析

为了更好地验证DHNR模型是否能够捕获时间信息，实验中采用了节点聚类进行测试．考虑到网络演化是一个漫长的过程，所以选择有较长时间的AMiner数据集进行实验．因为DHNR可以通过节点的所有时间轨迹来学习向量表示，时间步长越长，网络中包含的时间信息越多，节点的交互轨迹越丰富，学习到的节点向量表示越准确．本节通过分析不同的时间步长来验证本文模型在捕获网络演化方面的有效性．

实验结果如图6所示，当时间步长小于10时，节点聚类性能提升较快．这说明当时间步长较短时，网络中节点的交互轨迹不足以反应节点的特征信息．随着时间步长增加，包含的时间信息更加丰富，节点聚类精度提升较快．但是当时间步长达到一定值后，网络演化趋于稳定，模型已经能够从节点的交互轨迹中学习到较完整的节点向量表示，所以聚类结果趋于稳定．说明本

文模型能够真实地反映网络的动态演化特性.



(a) 不同时间步NMI值　　(b) 不同时间步ARI值

图6　不同时间步的影响

4.7　不同类型元路径性能分析

由于本文模型DHNR利用了多条不同类型的元路径来提取网络中的丰富语义信息,为验证单个不同元路径以及不同类型元路径的效果,以DBLP数据集为例,进行了节点聚类的实验验证.

如图7所示,仅仅使用单条元路径提取的语义信息和结构信息有限,导致性能较差. 而且从实验中可以发现,不同元路径提取的语义信息对节点表示向量的影响也不相同. APCPA较APA和APTPA具有更好的性能,分析原因是APCPA较好地反映了作者的研究领域与他们提交的会议之间的相关性. 由此可见,对不同类型的元路径进行注意力权重设置能够提升节点表示向量的质量.

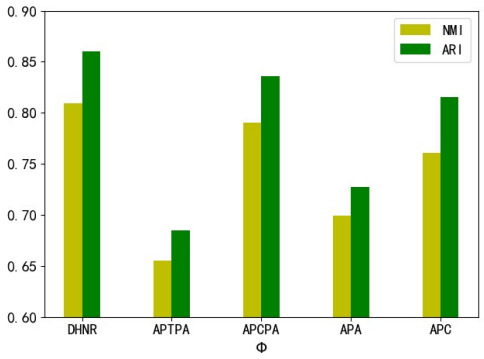
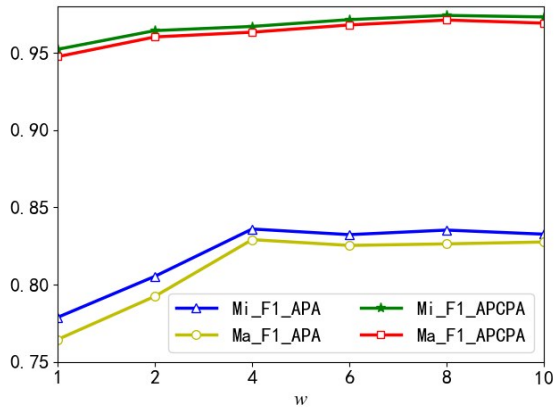


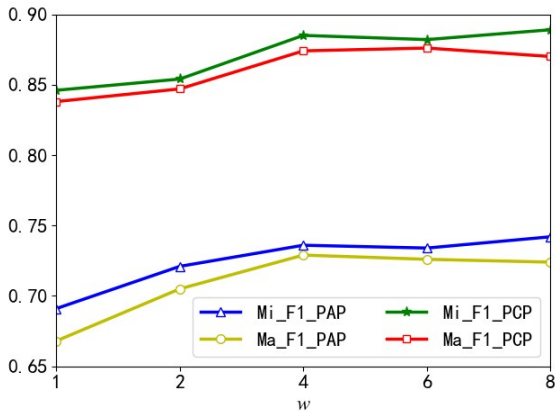
图7　不同元路径影响合

4.8　元路径数量分析

为判断不同元路径数量对模型的影响,对AMiner和DBLP-10数据集进行了节点分类的实验. 图8中Ma-F1表示Macro-F1,Mi-F1表示Micro-F1. 由图8(a)可知,在Aminer数据集上,APA元路径数量取为4的时候达到峰值,而APCPA元路径取值为8的时候达到峰值. 其中上下两组分别代表上述两条路径在不同数量情况下的Macro-F1和Micro-F1的取值情况. 由于在Aminer数据集中,很多作者发表论文数量是有限的,所以当路径数量为4的时候,就能够将APA的邻域空间准确划分,超过4则容易导致APA节点序列出现重复,再增加路径数量对模型几乎不产生影响. 对于APCPA,尽管一定数量序列可以将A的邻域空间准确划分,但是中继节点C的存在能够形成更多不同节点序列,因此更多的元路径数量能获取更多结构信息. 如图8(b)所示,在DBLP-10数据集上,当选取4条PAP路径和4条PCP元路径,实验效果达到峰值. 其中上下两组分别代表上述两条路径在不同数量情况下的Macro-F1和Micro-F1的取值情况. 随着路径数量增多,分类效果会逐渐提升. 但是到达一定值后,结果趋于稳定. 同样,在DBLP-4数据集上选取了4条APA,4条APCPA,4条APTPA进行对比实验.



(a) 在AMiner网络中不同路径数量实验结果



(b) 在DBLP-10网络中不同路径数量实验结果

图8　不同元路径数量实验结果

5　结论

本文提出了一种动态异质网络表示学习方法DHNR. 模型利用时间加权元路径和GRU模型来获取网络结构和语义信息,再通过带注意力机制的双向门控循环单元来归纳网络演化规律,有效地提高了节点向量表示的质量,并在分类、聚类等下游任务中均表现出优异的性能. 但是本文利用元路径提取语义信息可

能导致部分信息丢失,并且忽视了网络演化的驱动力.未来将从上述两个方面开展进一步的工作.首先是考虑采用能够更好地捕捉语义的方法,其次是从网络动力学角度更好地描述网络的动态演化过程.

参考文献

[1] JI M, HAN J W, DANIEVSKY M. Ranking-based classification of heterogeneous information networks[C]//Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego: ACM, 2011: 1298-1306.

[2] OPSAHL T, PANZARAS P. Clustering in weighted networks[J]. Social Networks, 2009, 31(2): 155-163.

[3] DONG Y, CHAWLA N V, SWAMI A. Metapath2vec: Scalable representation learning for heterogeneous networks[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Halifax: ACM, 2017: 135-144.

[4] 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, 2014: 701-710.

[5] 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. San Francisco: ACM, 2016: 855-864.

[6] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[J]. Advances in Neural Information Processing Systems, 2013, 26: 3111-3119.

[7] 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. Florence: ACM, 2015: 1067-1077.

[8] 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, 2016: 1225-1234.

[9] GORI M, MONFARDINI G, SCARSELLI F. A new model for learning in graph domains[C]//Proceedings of 2005 IEEE International Joint Conference on Neural Networks. Montreal: IEEE, 2005: 729-734.

[10] WANG X, JI H, SHI C, et al. Heterogeneous graph attention network[C]//The World Wide Web Conference. San Francisco: WWW, 2019: 2022-2032.

[11] QIAO Z, WANG P, FU Y, et al. Tree structure-aware graph representation learning via integrated hierarchical aggregation and relational metric learning[C]//20th IEEE International Conference on Data Mining. Sorrento: IEEE, 2020: 432-441.

[12] ZHANG C, SONG D, HUANG C, et al. Heterogeneous graph neural network[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage: ACM, 2019: 793-803.

[13] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[EB/OL]. (2017-02-22)[2021-03-01]. https://arxiv.org/pdf/1609.02907.pdf.

[14] ZHOU L, YANG Y, REN X, et al. Dynamic network embedding by modeling triadic closure process[C]//Proceedings of the AAAI Conference on Artificial Intelligence. New Orleans: AAAI, 2018, 32(1): 571-578.

[15] 尹赢, 张建朋, 吉立新, 李治成. 基于霍克斯点过程的动态网络表示学习方法[J]. 电子学报, 2020, 48(11): 2154-2161.
YIN Y, ZHANG J P, JI L X, Li Z C. Dynamic network representation learning based onhawkes point process[J]. Acta Electronica Sinica, 2020, 48(11): 2154-2161. (in Chinese)

[16] NGUYEN G H, LEE J B, ROSSI R A, et al. Continuous-time dynamic network embeddings[C]//Companion Proceedings of the The Web Conference 2018. Lyon: WWW, 2018: 969-976.

[17] LU Y, WANG X, SHI C, et al. Temporal network embedding with micro-and macro-dynamics[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing: ACM, 2019: 469-478.

[18] YIN Y, JI L X, ZHANG J P, et al. Dhne: Network representation learning method for dynamic heterogeneous networks[J]. IEEE Access, 2019: 134782-134792.

[19] WANG X, LU Y, SHI C, et al. Dynamic heterogeneous information network embedding with meta-path based proximity[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 34(3): 1117-1132.

[20] XUE H, YANG L, JIANG W, et al. Modeling dynamic heterogeneous network for link prediction using hierarchical attention with temporal RNN[C]//Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Ghent: Springer, 2020: 282-298.

[21] SUN Y, HAN J, YAN X, et al. Pathsim: Meta path-based top-k similarity search in heterogeneous information net-

works[J]. Proceedings of the VLDB Endowment, 2011, 4 (11): 992-1003.

[22] SHI C, ZHANG Z, LUO P, et al. Semantic path based personalized recommendation on weighted heterogeneous information networks[C]//Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. Shanghai: ACM, 2015: 453-462.

[23] HU Z, DONG Y, WANG K, et al. Heterogeneous graph transformer[C]//Proceedings of the Web Conference 2020. Taipei: ACM, 2020: 2704-2710.

[24] LE Q, MIKOLOV T. Distributed representations of sentences and documents[C]//International Conference on Machine Learning. Beijing: ACM, 2014: 1188-1196.

作者简介

刘　群　女,1969年出生于江西省南昌市,博士,重庆邮电大学教授,博士生导师. 主要研究方向为复杂网络、可解释人工智能和数据挖掘.
E-mail：liuqun@cqupt.edu.cn

谭洪胜　男,1996年出生于重庆市. 硕士生,主要研究方向为网络表示学习.
E-mail：redtan2000@163.com

张优敏　女,1987年出生于陕西省兴平市. 博士生,主要研究方向为图神经网络的可解释.
E-mail：ymzhang0103@hotmail.com

王国胤　男,1970年出生于重庆市,博士,重庆邮电大学教授,博士生导师. 教育部“长江学者”特聘教授(2015–2019)、中组部“万人计划”科技创新领军人才(2014)、人社部“新世纪百千万人才工程”国家级人选、国务院特殊津贴专家、中科院“百人计划”专家、教育部“新世纪优秀人才”. 主要研究方向为粗糙集、粒计算、数据挖掘、认知计算、大数据、人工智能等.
E-mail：wanggy@cqupt.edu.cn