中国科学: 信息科学 2020 年 第 50 卷 第 8 期: 1197–1216
SCIENTIA SINICA Informationis
论文

基于网络嵌入方法的耦合网络节点表示学习

韩忠明[1,2], 刘聃[1], 郑晨烨[1], 刘雯[1], 段大高[1*], 董健[3]

1. 北京工商大学计算机与信息工程学院, 北京 100048
2. 食品安全大数据技术北京市重点实验室, 北京 100048
3. 公安部第三研究所信息网络安全公安部重点实验室, 上海 200031
* 通信作者. E-mail: duandg@th.btbu.edu.cn

收稿日期: 2018–07–15; 修回日期: 2019–03–05; 接受日期: 2019–09–05; 网络出版日期: 2020–08–06

国家自然科学基金 (批准号: 61170112)、北京市自然科学基金 (批准号: 4172016)、北京市教委科技计划一般项目 (批准号: KM201710011006) 和公安部重点实验室开放课题资助项目

摘要 网络节点表示学习是网络数据分析挖掘中的一个基础问题, 通过学习网络节点表示向量, 可以更加精准地对网络节点进行表征. 近年来, 随着深度学习的发展, 嵌入方法在网络节点表示学习方面得到了广泛应用. 同时, 网络数据在规模、模态等特征方面也有了很大的变化, 研究重点从单网络分析挖掘逐渐演变至耦合网络分析挖掘. 本文首先分析了嵌入方法在单网络节点表示学习中的研究现状, 对比了现有方法的优劣. 然后借鉴单网络中嵌入方法的思想, 针对耦合网络提出了耦合网络嵌入模型 CWCNE. 针对耦合网络的特性, 改进了嵌入方法中的游走算法, 提出了一种网络间带约束的随机游走策略; 同时改进了模型的训练方法, 使用网络间迭代训练的方式来学习模型参数. 最后使用了社交耦合网络、学术耦合网络、影视耦合网络、诗词耦合网络、著作耦合网络等 5 组数据集验证了 CWCNE 的有效性. 并在社团划分、实体识别、标签分类等任务上取得了良好的结果.

关键词 网络嵌入, 节点向量, 耦合网络, 表示学习, 社团划分, 主体识别, 标签分类

1 引言

随着互联网的发展, 各式各样的网络数据不断产生. 这些网络数据之间往往不是独立的, 而是彼此之间有所联系, 形成了网络间的耦合关系. 以社交网络为例, 很多人不仅使用微信平台进行社交, 同时也使用微博、QQ 等平台进行交流. 同一用户实体在不同网络间有多个不同的账号, 这些不同账号即为耦合网络中的耦合节点. 在影视耦合网络中, 耦合现象更为明显, 如图 1 所示. 邓超作为电视剧演员出演了《相爱十年》《你是我的兄弟》等电视剧, 同时他作为电影演员出演了《美人鱼》《从你的全世界路过》等电影. 很多演员仅出现在电视剧网络或者电影网络中, 但是邓超同时出现在两个网络中, 就形成了网络间的耦合关系. 通过分析邓超在电视剧网络中的合作演员, 可以在电影网络中进行相关演

引用格式: 韩忠明, 刘聃, 郑晨烨, 等. 基于网络嵌入方法的耦合网络节点表示学习. 中国科学: 信息科学, 2020, 50: 1197–1216, doi: 10.1360/N112018-00182
Han Z M, Liu D, Zheng C Y, et al. Coupling network vertex representation learning based on network embedding method (in Chinese). Sci Sin Inform, 2020, 50: 1197–1216, doi: 10.1360/N112018-00182

© 2020《中国科学》杂志社

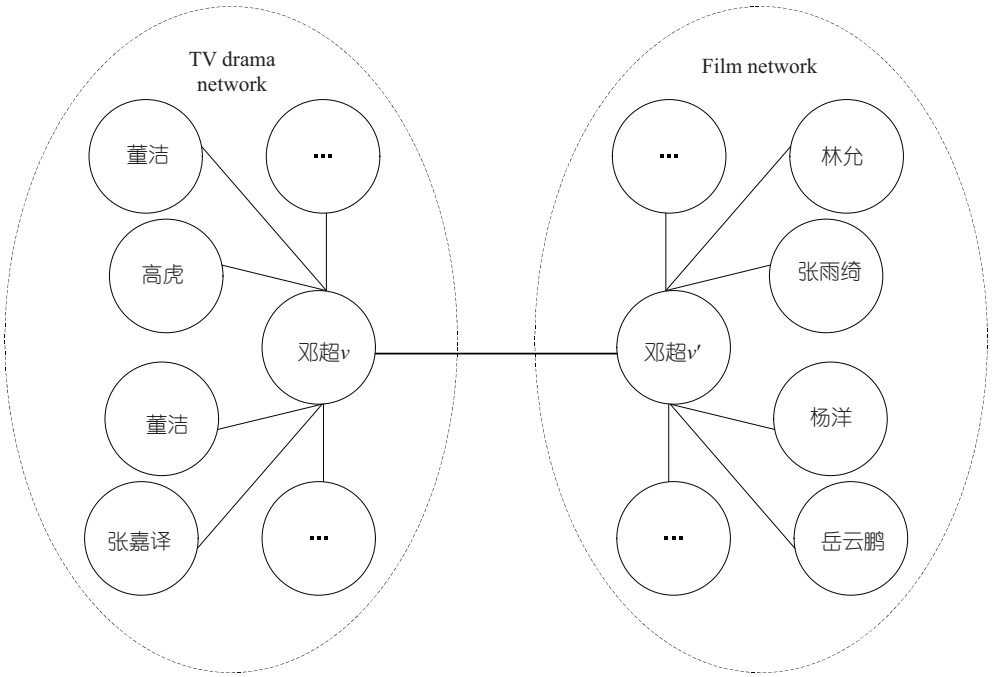

图 1　电视剧 – 电影耦合网络

Figure 1　TV drama – film coupling network

员的推荐. 研究耦合网络中的节点表示, 可以更好地刻画同一用户实体在不同网络中的属性、联系及行为等信息, 从而进行更精准的用户建模、个性化推荐等工作. 耦合网络的理论价值和应用价值不言而喻.

目前对于耦合网络的研究还处于起步阶段. 类比普通的单网络, 耦合网络中也存在节点表示 [1]、结构分析 [2]、消息传播 [3~7] 等问题. 针对耦合网络的耦合特性 [8], 还需要研究耦合度量 [9,10]、主体识别 [11]、耦合社团划分 [12] 等问题. 在这些问题中, 耦合网络节点表示是最为基础的一项研究.

近年来, 随着深度学习的发展, 嵌入类方法受到了广泛关注. 从最初的词嵌入 [13,14]、文本嵌入 [15]、衍生出网络嵌入 [16], 为网络节点表示学习提供了新的思路. 在此之前, 网络节点表示学习主要基于非负矩阵分解等方法 [17], 但此类方法所保留的网络节点信息较少, 在实际应用中的效果远不如嵌入类方法. 网络嵌入的基本思想与文本嵌入一致, 但由于网络拓扑结构是一种无序结构, 所以首先需要生成出合理的节点序列, 才能将网络嵌入问题转换为文本嵌入问题. 不同网络嵌入方法中的优化目标也略有不同.

耦合网络是通过多个具有少量耦合节点的网络作为桥梁连接而成的. 耦合网络上的表示学习是通过表示学习将耦合网络空间中的耦合节点及其邻域结构特性嵌入到表示空间中, 且表示相近的节点具有相似的邻域结构. 现有的网络节点表示学习方法大都只适用于单一网络, 对耦合网络节点表示学习的研究较少. 耦合网络中两个耦合网络往往结构差异较大, 所以需要修改序列生成算法和优化目标, 以适应耦合网络的复杂场景. 本文借鉴 DeepWalk 算法思想, 针对耦合网络提出了耦合网络嵌入模型 CWCNE. CWCNE 结合耦合网络特性, 设计了网络间带约束的随机游走策略, 不仅可以在耦合网络间学习到耦合节点的邻域信息, 还可以捕捉到非耦合节点之间相互影响的关系. 同时, CWCNE 改进了 DeepWalk 中模型学习方法, 实现网络间迭代的参数学习.

2 研究现状

近年来, 国内外学者对网络节点表示学习进行了大量研究, 随着各种网络的演化与结合, 对耦合网络节点向量化的研究也逐渐受到重视. 现有网络节点表示学习研究方法主要可以分为 4 大类, 第 1 类是基于非负矩阵分解的方法生成网络节点向量, 第 2 类是基于网络嵌入的方法生成网络节点向量, 第 3 类是基于深度学习框架生成网络节点向量, 第 4 类是多网络表示学习.

2.1 基于非负矩阵分解的网络表示学习方法

非负矩阵分解是从常规矩阵分解发展而来. 常规的矩阵分解包括主成分分析 (principal components analysis, PCA)、独立成分分析 (independent component analysis, ICA)、奇异值分解 (singular value decomposition, SVD)、矢量量化 (vector quantization, VQ) 等. 在这些方法中, 原始的大矩阵被近似分解为低秩的 $V = WH$ 形式. 这些方法的共同特点是, 因子 W 和 H 中的元素可为正或负, 即使输入的初始矩阵元素是全正的, 传统的矩阵分解算法也无法保证分解后矩阵的非负性. 在数学上, 从计算的观点看, 分解结果中存在负值是正确的, 但负值元素在实际问题中往往是没有意义的.

为了解决负值问题, 使矩阵分解的结果更好地得以应用, Lee 和 Seung 于 1999 年提出了一种新的矩阵分解思想 —— 非负矩阵分解 [18] (non-negative matrix factorization, NMF), 即在矩阵中所有元素均为非负数约束条件之下的矩阵分解方法. NMF 形式化定义为给定矩阵 $V \in \mathbb{R}^{n\times m}$, 寻找非负矩阵 $W \in \mathbb{R}^{n\times r}$ 和非负矩阵 $H \in \mathbb{R}^{r\times m}$, 使得 $V \approx WH$. 原始矩阵的列向量是对左矩阵中所有列向量的加权和, 而权重系数就是右矩阵对应列向量的元素, 故称 W 为基矩阵, H 为系数矩阵. 一般情况下 r 的选择要比 n 小, 即满足 $(m+n)r < mn$, 这时用系数矩阵代替原始矩阵, 就可以实现对原始矩阵进行降维, 从而减少存储空间和计算资源. 其中系数矩阵 H 由多个低维向量构成, 当为网络邻接矩阵时, H 的行向量即为网络节点的低维表示.

常用的非负矩阵分解方法包括局部非负矩阵分解 [19] (local non-negative matrix factorization, LNMF)、加权非负矩阵分解 [20] (weighted non-negative matrix factorization, WNMF)、Fisher 非负矩阵分解 [21] (fisher non-negative matrix factorization, FNMF)、稀疏非负矩阵分解 [22] (sparse non-negative matrix factorization, SNMF)、受限非负矩阵分解 [23] (constrained non-negative matrix factorization, CNMF)、非平滑非负矩阵分解 [24] (non-smooth non-negative matrix factorization, nsNMF) 等. 各种非负矩阵分解方法之间主要差别在于所要优化的目标函数不同. 常用的目标函数包括欧式距离、KL 散度、高阶范数等.

2.2 基于网络嵌入的网络表示学习方法

网络嵌入由传统的词嵌入/文档嵌入衍生而来, 但由于网络结构不同于文本结构, 所以在进行网络嵌入操作前, 首先需要解决这些异同点. 在对文本进行向量化时, 文本通常是单向且有序的, 可以通过有序的文本, 获取上下文序列, 从而使用 CBOW, SkipGram 等模型进行建模. 但在网络中, 通常存在大量的节点和边, 这些点和边构成的网络拓扑结构要比文本结构复杂许多. 所以在生成网络节点上下文序列时, 需要很多专门的方法来构造节点序列.

最早的方式是 DeepWalk, 于 2014 年提出, 该模型首先解决了节点序列生成的问题, 使用随机游走的方式产生节点序列, 对应词嵌入中的上下文序列. Tang 等 [25] 于 2015 年提出的 LINE 模型, 提出了另一种生成节点序列的方法, 考虑了网络中的一阶和二阶相似性, 并使用边采样的方式对算法进行了优化, 也取得了不错的网络节点表示向量. Grover 等 [26] 提出了 node2vec 模型, 对 DeepWalk 中的

随机游走算法进行了改进, 并加入了更多的采样策略, 是目前学者们常用的一种网络嵌入生成方法.

上述几种方法均可以用来生成网络节点表示向量, 但都存在同一个问题, 就是仅包含局部网络信息, 而忽略了全局网络信息. 为了解决该问题, GENE 模型 [27] 借鉴了文档嵌入中文档向量的思路, 引入了组向量的概念, 将网络中的群组信息/全局信息引入到网络节点序列中, 使得新的网络节点表示向量不仅包含局部信息, 同时也包含网络的全局信息.

2.3 基于深度学习框架的网络表示学习方法

近年来深度学习在计算机视觉、语言建模等多种研究领域显示出了突出的性能. 基于深度学习框架的网络表示学习方法主要关注如何使深度学习模型适应网络结构数据, 并对模型损失增加网络结构和属性的约束 [28]. SDNE 模型 [29] 首次将深度学习模型应用于网络表示学习, 该模型使用拉普拉斯 (Laplace) 矩阵监督一阶相似度建模, 由无监督的深层自编码器对二阶相似度建模, 将深层自编码器的中间层作为节点表示, 取得了很好的网络节点表示向量. 此外, 一些方法使用卷积神经网络进行节点嵌入学习, 如 Kipf 等 [30] 提出的 GCN 模型, 该模型针对节点分类问题构建了半监督的节点嵌入模型对网络拓扑结构和网络节点特征进行编码, 从而为网络习得节点表示. Hamilton 等 [31] 提出的 GraphSAGE 模型使用聚合函数来定义图上的卷积, 是一种归纳学习方法, 通过对节点的邻域信息进行采样、聚合, 快速地生成节点嵌入. 同样, Chen 等 [32] 提出的 FastGCN 模型, 将图卷积解释为嵌入函数在概率度量下的积分变换, 采用蒙特卡洛 (Monte Carlo) 方法估计积分, 同时使用重要性采样, 提升了训练效率. 此外, Pan 等 [33] 提出的 ARGA 模型基于自动编码器 [34] (autoencoder, AE) 进行结点表示学习, 结合重构误差和隐变量分布与先验分布的误差构建损失函数, 使用对抗训练方法在 AE 的解码器后引入判别器, 并加入正则化项以增加模型的鲁棒性. Wang 等 [35] 提出的 GraphGAN 模型使用生成对抗式网络 [36] (generative adversarial networks, GAN) 对结点相连的概率进行建模, 通过对抗训练交替优化生成器和判别器, 使生成器学习到的表示向量可以包括图的局部拓扑结构. 并且, GraphGAN 采用 graph softmax 的方法解决了 softmax 函数复杂度过高的问题.

上述网络表示学习方法均在单网络分析挖掘中取得了不错的效果, 但耦合网络在属性、关系、结构等方面均与单网络有较大差异. 针对耦合网络, 需要提出更加有效的网络表示学习方法.

2.4 多网络的网络表示学习

现有的多网络表示学习方法主要针对多关系型网络, 即网络节点之间有多类不同含义的边. Matsuno 等 [14] 认为网络由多层节点相同, 而关系类型不同的网络组成, 并针对这种多层复合网络提出了网络嵌入算法 MELL. 该算法分别对每层网络习得嵌入, 并提出构建层向量用来捕获和表征不同层网络之间的连通性, 使得不同层之间的节点嵌入尽可能相近. Qu 等 [37] 将同一网络节点间边的不同类型作为网络的多个视图, 即按照网络边的种类构建多个节点相同边种类不同的视图, 并提出了多视图网络嵌入算法 MVE (multi-view network embedding). MVE 分别对网络的多个视图进行嵌入, 通过添加注意力的方法得到节点不同视图的权重, 最终的网络节点表示为多个视图嵌入的加权和. Xu 等 [38] 针对异质网络的多网络学习提出了 EOE (embedding of embedding) 算法, EOE 对网络中的边进行了嵌入, 不仅对单个网络中的边进行嵌入, 还通过构造均衡嵌入矩阵进一步对多个异构网络之间的边进行编码. 实验证明上述多网络嵌入方法均优于单网络嵌入方法. 耦合网络表示学习与上述多网络分析研究类似, 即均对多个网络中的信息进行学习, 然而他们的具体研究对象不同. 与普通多网络分析研究相比, 耦合网络的研究对象是具有少量节点作为耦合节点的多个网络, 且耦合节点邻域结构重叠. 而多元网络分析的研究对象是通过大量相同节点而关联起来的网络, 且这些网络相同节点间的边种类可能不

同. 因此, 耦合网络是通过少量耦合节点连接的多个网络, 通过耦合节点可以将多个网络连接成为耦合网络, 通过对耦合网络进行学习, 可以更好地捕捉到节点邻域结构特征.

3 相关定义

节点 (vertex): 节点是网络中的一个功能个体, 如社交网络中的一名用户, 学术网络中的一名学者, 影视网络中的一名影人, 词共现网络中的一个词语, 著作网络中的一名作者等, 都可以看作一个节点. 用 V 表示节点构成的集合, 用 v_i 表示一个具体的节点.

边 (edge): 边是用来刻画两个节点之间关系的, 可能具有方向性, 称为有向边, 如社交网络中的单向关注关系; 也可能不具备方向性, 称为无向边, 如词共现网络中的词语共现关系. 本文实验中的边均为无向边. 用 E 表示边构成的集合, 用 e_{ij} 表示一条具体的边.

网络 (network): 网络是对关系数据的刻画, 定义网络 $G=(V,E)$, 其中 V 是 G 中的节点集合, E 是 G 中的边集合.

定义1 (耦合网络 (coupling network)) 对于两个独立的网络 G_1, G_2, 若两个网络中存在对应相同实体的节点对 $\langle v_i^1, v_j^2\rangle$, 则称 v_i^1, v_j^2 为耦合节点, 称 $G=[G_1,G_2]$ 构成的新网络为耦合网络.

以图 1 中的影视耦合网络为例. 在电视剧网络 G_1 中, 邓超、董洁、高虎、张嘉译等演员均为网络中的节点 v_i^1, 构成了节点集合 V^1, 演员由于共同出演过同一部电视剧而形成了共演关系 e_{ij}^1, 构成了边集合 E^1, 同理, 电影网络 G_2 中也存在类似的节点集合 V^2 和边集合 E^2, 电视剧网络 G_1 和电影网络 G_2 共同构成了影视耦合网络 G. 邓超由于同时出现在两个网络中, 所以是耦合节点.

网络表示学习是对给定的网络 $G=(V,E)$, 学习一个模型 $f(v)$, 使得对任意一个节点 v, $f(v)\rightarrow r\in\mathbb{R}^d$, r 是一个稠密的实向量, 并且 $d<|V|$. 根据网络表示学习, 我们可以定义耦合网络表示学习.

定义2 (耦合网络节点表示学习 (coupling network node representation learning)) 对于由 G_1, G_2 构成的耦合网络中的每个节点 v_i^1, v_j^2, 学习一组低维向量 $\phi_v\in\mathbb{R}^d$, ϕ_v 是一个维度为 d 的低维稠密实数向量, 并且满足 $d\ll|V_1V_2|$. 其中 $|V_1V_2|$ 表示两个网络的节点总数. 由 $|V_1V_2|$ 个 d 维向量构成的矩阵 $\Phi\in\mathbb{R}^{|V_1V_2|\times d}$ 为耦合网络节点的表征矩阵. 学习矩阵 Φ 的过程则为耦合网络节点表示学习过程.

对图 1 中的耦合网络实例而言, 耦合网络节点表示学习是构建一个学习方法 M, 可以为每个节点生成一个节点向量 ϕ_v, 该向量同时包含了演员在两个网络中的信息.

4 耦合网络表示学习模型 CWCNE

耦合网络节点表示学习, 旨在从耦合网络数据中学习到耦合网络节点表示向量. 无论是嵌入式表示学习方法还是矩阵分解的表示学习方法, 都需要得到节点的上下文特征, 如局部或全局的序列游走等. 对于耦合网络, 节点间存在耦合关联, 如何在表示学习的特征中利用并保持耦合关联成为基础问题. 本文根据耦合网络的特征, 提出一个交叉随机游走生成上下文特征的表示学习方法 (crossing randorm walking for coupling network embedding, CWCNE).

4.1 CWCNE 模型

CWCNE 是嵌入耦合网络表示学习方法. 为了在表示学习的特征中利用并保持耦合关联, 我们在节点随机游走的过程中引入了跨网络的约束随机游走, 使得训练一个网络的节点向量时, 引入了耦合网络关联节点的特征信息, 耦合网络的学习示意图如图 2 所示. 图 2 中, 训练节点 V 的向量时, 我们

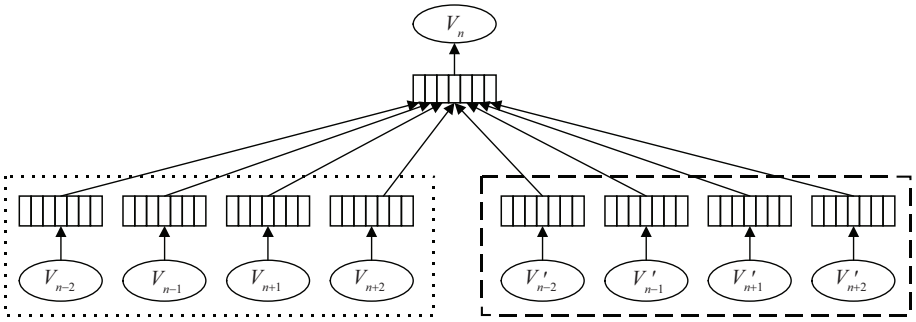

图 2 CWCNE 模型

Figure 2 Model: CWCNE

将节点 V 的耦合关联的节点 V' 及其上下文信息作为正则化引入, 这样在节点 V 的向量中保持了耦合关联的特性.

在网络嵌入表示学习中, 首先需要通过随机游走策略生成节点序列 W_{v_i}, 其中 ω 为随机游走的窗口大小, 然后使用 W_{v_i} 作为嵌入方法 SkipGram 的输入, 则模型的目标函数为

$$\text{minimize}_{\Phi} -\log(\Pr(\Phi v_{i-\omega},\ldots,v_{i+\omega}|\Phi(v_i))\Pr(\Phi v'_{i-\omega},\ldots,v'_{i+\omega}|\Phi(v_i))). \tag{1}$$

由于 SkipGram 不考虑序列的顺序, 所以式 (1) 的条件概率之间相互独立, 则有

$$\Pr(\Phi v_{i-\omega},\ldots,v_{i+\omega}|\Phi(v_i))\Pr(\Phi v'_{i-\omega},\ldots,v'_{i+\omega}|\Phi(v_i)) = \prod_{i-\omega,j\neq i}^{i+\omega}\Pr(\Phi(v_j|\Phi(v_i)))\Pr(\Phi(v'_j|\Phi(v_i))), \tag{2}$$

其中, 计算 $\Pr(\Phi(v_j|\Phi(v_i)))$ 及 $\Pr(\Phi(v'_j|\Phi(v_i)))$ 的计算量非常大, 因此使用 Hierarchical Softmax 方法来解决这一问题, 采用 Hierarchical Softmax 方法还能保证参数学习过程可以较快地收敛.

假设网络中的每一个节点对应于 Huffman 树的一个叶子节点, 可将网络中所有节点的线性概率最大化问题转化为 Huffman 树由根节点到某一叶子节点的概率最大化的问题. 假设节点 u_k 为 Huffman 树的一个叶子节点, 从 Huffman 树的根节点到叶子节点 v_j 所有的非叶子节点为 $(b_0,b_1,\ldots,b_{\lceil\log|V|\rceil-1})$, 其中 b_0 为根节点, $b_{\lceil\log|V|\rceil-1}$ 为 v_j, v_j 的耦合节点 v'_j 在学习过程中会与其共享二叉树, 得到

$$\Pr(\Phi(v_j|\Phi(v_i)))\Pr(\Phi(v'_j|\Phi(v_i))) = \prod_{l=1}^{|\log|V||}\Pr(\phi(b_l)|\Phi(v_j))\Pr(\phi(b_l)|\Phi(v'_j)), \tag{3}$$

通过引入 Huffman 树表示 $\Pr(\Phi(u_k)|\Phi(v_j))$, 将其转化成了 $\lceil\log|V|\rceil$ 个二分类, 二分类函数使用 logistic 分类函数, 得到

$$\Pr(\phi(b_l)|\Phi(v_j))\Pr(\phi(b_l)|\Phi(v'_j)) = \frac{1}{1+\mathrm{e}^{-\Phi(v_j)\phi(b_l)}}\frac{1}{1+\mathrm{e}^{-\Phi(v'_j)\phi(b_l)}}, \tag{4}$$

其中 b_l 为非叶子节点, $\Phi(b_l)\in\mathbb{R}^d$ 即非叶子节点的向量映射函数, $\Phi(b_l)$ 为类别向量. 通过 Hierarchical Softmax 方法我们将计算 $\Pr(\Phi(v_j|\Phi(v_i)))$ 及 $\Pr(\Phi(v'_j|\Phi(v_i)))$ 的时间复杂度由 $O(|V|)$ 降低为 $O(\lceil\log|V|\rceil)$. 模型参数为 $\theta=\Phi,\phi$, 参数可以通过随机梯度下降 (stochastic gradient descent, SGD) 进行参数的学习. 为了使用 SGD, 需要求出参数的梯度. 将式 (4) 写成整体的形式, 有

$$\begin{aligned}&\Pr(b_l|\Phi(v_j),\phi^j_{l-1})\Pr(b_l|\Phi(v'_j),\phi^{j'}_{l-1})\\&\quad=[\sigma\Phi(v_j)^{\mathrm{T}}\phi^j_{l-1}(b_l)]^{1-b_l}[1-\sigma\Phi(v_j)^{\mathrm{T}}\phi^j_{l-1}(b_l)]^{b_l}[\sigma\Phi(v'_j)^{\mathrm{T}}\phi^{j'}_{l-1}(b_l)]^{1-b_l}[1-\sigma\Phi(v'_j)^{\mathrm{T}}\phi^{j'}_{l-1}(b_l)]^{b_l},\end{aligned} \tag{5}$$

将式 (5) 代入式 (3), 再将式 (3) 代入式 (2), 对式 (2) 求对数似然可得

$$L=\sum_{i\in V}\sum_{j=i-\omega,j\neq i}^{i+\omega}\sum_{i=1}^{\lceil\log|V|\rceil}\left\{\begin{array}{l}(1-b_l)[\sigma\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l)]\\ +b_l[1-\sigma\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l)]\\ +(1-b_l)[\sigma\Phi(v_j')^{\mathrm{T}}\phi_{l-1}^{j'}(b_l)]\\ +b_l[1-\sigma\Phi(v_j')^{\mathrm{T}}\phi_{l-1}^{j'}(b_l)]\end{array}\right\},\tag{6}$$

令

$$\begin{aligned}L(v_j,j,v_j')=&(1-b_l)[\sigma\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l)]+b_l[1-\sigma\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l)]\\&+(1-b_l)[\sigma\Phi(v_j')^{\mathrm{T}}\phi_{l-1}^{j'}(b_l)]+b_l[1-\sigma\Phi(v_j')^{\mathrm{T}}\phi_{l-1}^{j'}(b_l)],\end{aligned}\tag{7}$$

分别对式 (7) 求关于 $\Phi(v_j)$, $\Phi(v_j')$ 和 $\phi_{l-1}^{j}(b_l)$ 的偏导数:

$$\begin{aligned}\frac{\partial L(v,j,l)}{\partial\Phi(v_j)}&=[1-b_l-\sigma(\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l))]\phi_{l-1}^{j}(b_l),\\ \frac{\partial L(v,j,l)}{\partial\Phi(v_j')}&=[1-b_l-\sigma(\Phi(v_j')^{\mathrm{T}}\phi_{l-1}^{j'}(b_l))]\phi_{l-1}^{j'}(b_l),\\ \frac{\partial L(v,j,l)}{\partial\phi_{l-1}^{j}(b_l))}&=[1-b_l-\sigma(\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l))]\Phi(v_j),\end{aligned}\tag{8}$$

由式 (8) 通过梯度下降法可得

$$\begin{aligned}\Phi(v_j)&=\Phi(v_j)-\eta\sum_{j=i-\omega,j\neq i}^{i+\omega}\sum_{i=1}^{\lceil\log|V|\rceil}[1-b_l-\sigma(\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l))]\phi_{l-1}^{j}(b_l),\\ \Phi(v_j')&=Phi(v_j')-\eta\sum_{j=i-\omega,j\neq i}^{i+\omega}\sum_{i=1}^{\lceil\log|V|\rceil}[1-b_l-\sigma(\Phi(v_j')^{\mathrm{T}}\phi_{l-1}^{j'}(b_l))]\phi_{l-1}^{j'}(b_l),\\ \phi_{l-1}^{j}(b_l)&=\phi_{l-1}^{j}(b_l)-\eta[1-b_l-\sigma(\Phi(v_j)^{\mathrm{T}}\phi_{l-1}^{j}(b_l))]\Phi(v_j),\end{aligned}\tag{9}$$

如此便可求解出模型参数 $\theta=\Phi,\phi$.

下面我们分别介绍序列生成阶段的耦合约束随机游走算法以及模型求解算法.

4.2 耦合约束随机游走

由于耦合网络数据中同时包含两个或多个网络, 所以不能简单使用单网络中的随机游走算法. 为了引入耦合网络间的节点信息, 需要对游走策略进行约束, 使得节点序列中不仅包含当前节点的上下文信息, 同时也要包含其对应的耦合节点的上下文信息.

算法 1 为耦合网络约束随机游走算法 (coupling random walk), 用于生成耦合节点序列.

模型输入共有 8 项. 由 G_1, G_2 构成的耦合网络, 耦合节点集合 CV, 耦合节点向量维度 d, 每个节点的游走次数 γ, 单次游走的最大步长 t, 耦合节点在耦合网络中游走的最大步长 t', 窗口大小 ω. 模型输出为耦合网络节点的表征矩阵 Φ.

在初始化阶段, 需要对表征矩阵 Φ 和二叉树 T 进行初始化 (第 1, 2 行). 算法主体部分, 对网络中的每个节点进行 γ 次游走 (第 3 行). 在每次游走开始前, 为节点生成一个随机序列, 这样可以加速随机梯度下降的训练过程 (第 4 行). 遍历随机序列中的每一个节点 (第 5 行), 若节点 v_i 不在耦合节点集合中, 则在当前网络中进行步长为 t 的随机游走 (第 6, 7 行), 若节点 v_i 在耦合节点集合中, 则

Algorithm 1 CouplingSkipWalk

Input: $G_1=(V,E), G_2=(V,E)$, coupling vertex CV, embedding size d, walks per vertex γ, walk length t, t', window size ω;

1: Initialization: Sample Φ from $\mathbb{R}^{|V_1V_2|\times d}$;
2: Build a binary Tree T from $|V_1V_2|$;
3: **for** $i=0$ to γ **do**
4: $O=\text{suffle}(|V_1V_2|)$;
5: **for** $v_i \in O$ **do**
6: **if** $v_i \notin$ CV **then**
7: $W_{v_i}=\text{RandomWalk}(G, v_i, t)$;
8: **end if**
9: **if** $v_i \in CV$ **then**
10: $W_{v_i}=\text{concat}(\text{RandomWalk}(G, v_i, t-t'), \text{RandomWalk}(G, v_i', v'))$;
11: **end if**
12: $\text{CouplingSkipGram}(\Phi, W_{v_i}, \omega, \text{CV})$;
13: **end for**
14: **end for**

Output: matrix of vertex representation $\Phi \in \mathbb{R}^{|V_1V_2|\times d}$.

在当前网络中进行步长为 $t-t'$ 的随机游走, 并在对应的耦合网络 G' 中为其对应的耦合节点 v_i' 进行步长为 t' 的随机游走 (第 8, 9 行) 生成节点序列 W_{v_i}. 将第 7, 9 行中产生的网络节点序列输入 CouplingSkipGram, 更新节点向量.

上述过程中, G_1, G_2 构成耦合网络, G 表示当前节点 v_i 所在的网络, G' 表示其耦合节点 v_i' 所在的网络. 二叉树 T 将作为参数输入到 CouplingSkipGram 算法中. 步长 t, t' 均为最大步长, 且 $t' \leqslant t$, 实际节点序列的长度可小于 t, 即 $|W_{v_i}| \leqslant t$, 对于后续的 CouplingSkipGram 算法, 节点序列也可是不定长的. 在随机游走过程中, 同一个节点不能重复出现.

4.3 模型求解算法

模型求解的主要任务是更新参数 T 及参数 Φ. 其中参数 T 为节点序列对应的二叉树, 是一个中间参数. 参数 Φ 为最终待求解的耦合网络表征矩阵. 算法 2 为 CouplingSkipGram 算法, 用于更新耦合节点向量. 对于输入的节点序列 W_{v_i}, 遍历其中的每一个节点 v_j (第 1 行). 对于当前 v_j, 遍历上下

Algorithm 2 CouplingSkipGram

1: **for** $v_j \in W_{v_i}$ **do**
2: **for** $u_k \in W_{v_i[j-\omega:j+\omega]}$ **do**
3: $J(\Phi)=-\text{logPr}(u_k|\Phi(v_j))$;
4: $\Phi=\Phi-\alpha\frac{\partial J}{\partial \Phi}$;
5: **if** $u_k \in$ CV **then**
6: $J(\Phi)=-\text{logPr}(u_k'|\Phi(v_j))$;
7: $\Phi=\Phi-\alpha\frac{\partial J}{\partial \Phi}$;
8: **end if**
9: **end for**
10: **end for**

文窗口 ω 内的所有节点 u_k (第 2 行). 对于每个 u_k, 计算目标函数 $J(\Phi)$ (第 3 行), 并使用随机梯度下降法更新表征矩阵 Φ (第 4 行). 若节点 u_k 在耦合节点集合中, 则在对应的耦合网络 G 中为其对应的

表 1 6 组耦合网络数据集的基本统计指标

Table 1 The basic statistical indicators of 6 sets of coupled network datasets

Dataset	Number of vertexes in network 1	Number of edges in network 1	Number of vertexes in network 2	Number of edges in network 2	Number of coupling vertexes
SCN	22	74	25	103	8
ACN	2985414	25965384	1053188	3916907	733592
FCN	9274	138065	2805	293848	1947
PCN	3429936	27519883	92385	687327	46260
WCN	372971	919276	17365	38247	15406
MCN	10000	20000	10000	10000	2000

耦合节点 u'_k 进行相同的操作 (第 6, 7 行).

5 实验结果与分析

5.1 数据集

为了客观全面地衡量本文模型的效果, 我们采用了不同的耦合网络进行分析. 目前尚无耦合网络的基准测试数据集, 所以我们构造了一个基准的社交耦合网络, 然后选择了具有代表性的学术耦合网络、影视耦合网络、诗词耦合网络、著作耦合网络等 5 组数据集. 表 1 给出了 6 组耦合网络数据集的基本统计指标.

社交耦合网络 (social coupling network, SCN): SCN 是我们人工采集的真实社交网络数据, 包含 QQ、微信两个社交网络. 其中 QQ 数据中包含 22 个用户和 74 条用户关系, 微信数据中包含 25 个用户和 103 个用户关系.

学术耦合网络 (academic coupling network, ACN): 来自文献 [30, 39] 中提出的学术网络, 其中 LinkedIn 是一个职业网络, 用户在此网络中展示个人信息并用于社交, 由于无法从 LinkedIn 网络中获取用户关系, 所以使用 co-view 关系来替代用户关系, 该网络数据中共包含 2985414 条用户信息和 25965384 条用户关系. ArnetMiner 网络提供了学术社区的学者搜索和挖掘服务, 本数据收集自 2013 年, 包含 1053188 条用户信息和 3916907 条用户关系.

影视耦合网络 (film coupling network, FCN): FCN 是我们从影视门户采集的网络, 包含中国内地 (大陆) 及港澳台地区的大部分电影、电视对应的影人数据. 使用影人的共演关系代替用户关系. 其中电影数据中包含 2805 条用户信息和 293848 条用户关系, 电视数据中包含 9274 条用户信息和 138065 条用户关系.

诗词耦合网络 (poetry coupling network, PCN): 来自网络, 包含唐代和宋代的大部分唐诗和宋词. 使用诗词中的词语共现关系代替词语关系. 其中唐诗数据中包含 92385 条词语信息和 687327 条词语关系, 宋词数据中包含 3429936 条词语信息和 27519883 条词语关系.

著作耦合网络 (work coupling network, WCN): 来自 dblp 数据库, 包含论文合作数据和书籍合作数据. 使用著作中的合作共现关系代替作者关系. 其中论文合作数据中包含 372971 条作者信息和 919276 条作者关系, 书籍合作数据中包含 17365 条作者信息和 38247 条作者关系.

人工耦合网络 (manual coupling network, MCN): 人工模拟生成的耦合网络. 其中网络 1 为小世界网络, 包含 10000 个节点及 20000 条边, 网络 2 由网络 1 复制而成, 但删减掉了 50% 的边, 所以包

含 10000 个节点及 10000 条边. 网络间选取 20% 的节点作为耦合节点.

数据准备: 在数据准备阶段, 将每组数据集中的两个网络组合成一个新的耦合网络, 其中耦合节点对中的两个节点视为不同的两个节点, 为其加入一条边, 表示两个节点间有联系. 在实体识别任务中, 将部分耦合节点对之间的边去掉, 作为测试数据.

5.2 对比方法

目前, 耦合网络的表示学习方法尚无公开报道, 为了客观比较和衡量本文模型的效果, 我们将耦合网络的耦合节点相连, 这样将耦合网络转化成单一网络, 然后选择目前具有代表性的两个网络表示学习方法作为对比, 如下所述.

DeepWalk: 该方法主要包括生成节点序列和更新模型参数两个阶段. 生成节点序列阶段, 使用随机游走方法, 更新模型参数阶段使用 SkipGram 方法, 其中使用层次 Softmax 提高学习效率, 使用随机梯度下降学习具体参数. 该方法是单网络中节点表示学习的常用方法. DeepWalk 中将随机游走的数量设定为 40, 随机游走的步长设定为 80, 语言模型窗口大小设定为 10, 最终表示向量的维度设为 128.

Node2vec: 该方法改进了随机游走策略, 增加了搜索偏置项, 可以通过设置搜索偏置项来调节游走的深度与广度, 使用随机梯度下降学习具体参数. 该方法可以尽量多地包含网络中的局部信息和全局信息. 由于 Node2vec 是在 DeepWalk 算法基础上改进随机游走策略, 因此本实验设置该算法的参数与 DeepWalk 算法参数相同.

LINE: 该方法定义每对顶点的两个联合概率分布, 一个使用邻接矩阵, 另一个使用嵌入, 通过最小化两个分布的 KL 散度得到包含网络一阶和二阶相似性的节点嵌入. 本实验使用二阶 LINE 算法, 输出表示向量维度设为 128.

SDNE: 该方法用带监督的 Laplace 矩阵对一阶相似度建模, 用无监督的深层自编码器对二阶相似度建模, 并将深层自编码器的中间层作为节点的网络表示. SDNE 中本实验使用两层深度编码器进行表示学习, 编码器隐藏层神经元个数设定为 256, 输出层神经元个数设定为 128, 即最终的网络表示维度是 128.

5.3 社团划分

在社交网络中, 用户相当于每一个节点, 用户之间通过互相的关注关系构成了整个网络的结构, 在这样的网络中, 有的用户之间的连接较为紧密, 有的用户之间的连接关系较为稀疏, 在这样的网络中, 连接较为紧密的部分可以被看成一个社团, 其内部的节点之间有较为紧密的连接, 而在两个社团间则相对连接较为稀疏, 社团划分旨在发现有紧密连接的社团结构 [40].

社团划分任务作为网络分析中的基础任务, 可以用来验证耦合网络节点向量的有效性. 使用模块度来衡量社团划分的优劣, 模块度越大, 则社团划分的效果越好, 其对应的耦合网络节点向量的有效性越强. 对于社团划分任务, 使用 K-means 方法对耦合网络节点表示向量进行聚类, 将同类别节点视作同一社团成员. 图 3 展示了 CWCNE 模型在社交耦合网络中的社团划分结果. 其中图 3(a) 为社交耦合网络的拓扑结构图, 图 3(b) 为节点向量降维后的 2 维表示. 耦合节点向量虽然包含了耦合网络的信息, 但是并不会影响网络间社团的划分, 从图 3(a) 中可以看出, 不同网络间的社团依旧独立. 耦合网络中的多个网络虽然存在很多耦合节点对, 但是耦合节点间的邻边个数远小于单一网络内部节点间的邻边个数, 所以不易形成耦合的社团结构.

图 4 对比了 5 种方法在影视耦合网络数据集中社团划分的结果, 可以发现, 在不同值下, CWCNE 均取得了最佳的划分效果. 当聚类中心在 $[4, 7]$ 范围内逐渐增大时, 模块度随着聚类中心数的增加而增

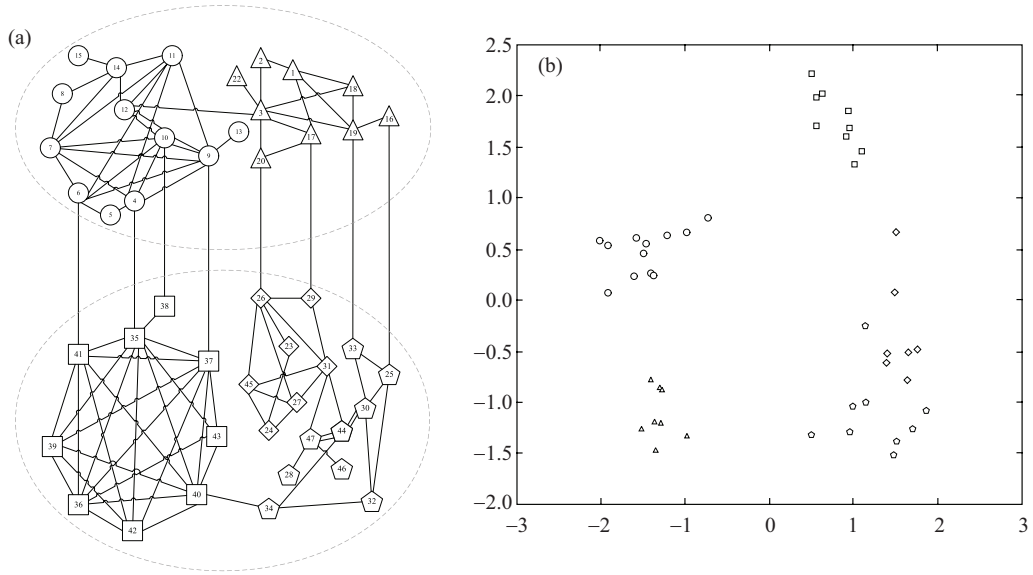

图 3 社交耦合网络数据集中社团划分结果. (a) 社交耦合网络拓扑结构图; (b) 节点向量可视化结果图

Figure 3 Results of community detection in social coupling network. (a) Topology of social coupling network; (b) visualization of node representation vectors

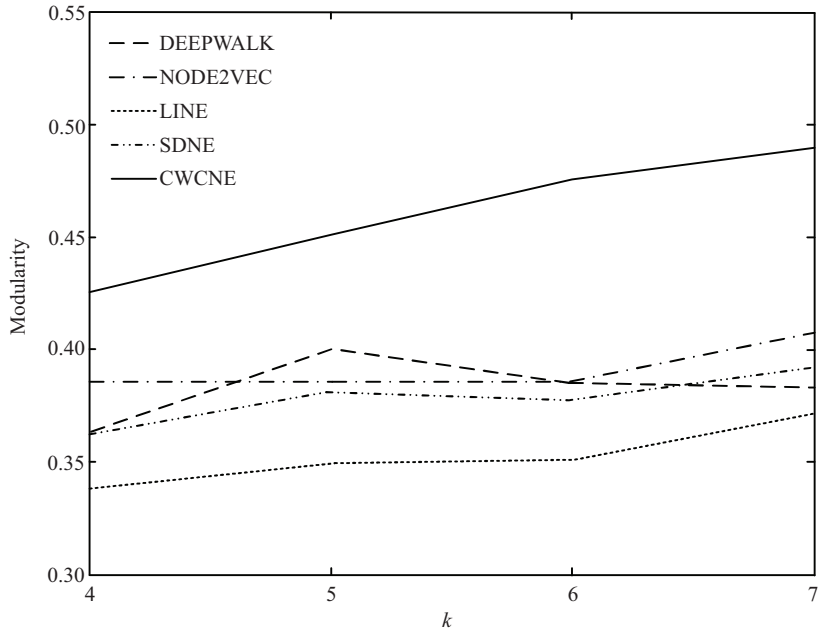

图 4 影视耦合网络数据集上各算法在不同值下的模块度对比

Figure 4 Modularity contrast of algorithms with different k on film coupling network

大. 其中 deepwalk, node2vec 等对比算法对聚类中心数不太敏感, CWCNE 的模块度随着聚类中心数的增长而明显增大. 在 $k = 7$ 时取得最大的模块度. 在 $k = 7$ 时, CWCNE 模型对应的模块度为 0.49, node2vec 的模块度为 0.40, deepwalk 的模块度为 0.38, SDNE 的模块度为 0.39, LINE 的模块度为 0.37, CWCNE 在社团划分任务中有明显优势. CWCNE 保留了更多的耦合网络社团结构信息.

表 2　不同数据集下耦合网络社团划分模块度

Table 2　Comparison of module degree of community detection in coupled network

Dataset	Deepwalk	Node2vec	LINE	SDNE	CWCNE
SCN ($k=5$)	0.334	0.345	0.274	0.312	0.381
ACN ($k=8$)	0.396	0.431	0.337	0.423	0.426
FCN ($k=7$)	0.383	0.408	0.372	0.392	0.490
PCN ($k=13$)	0.435	0.443	0.391	0.413	0.391
WCN ($k=8$)	0.351	0.385	0.349	0.391	0.373
MCN ($k=46$)	0.524	0.507	0.493	0.537	0.598

表 3　不同数据集下, 单网络社团划分模块度

Table 3　Module degree of single network community detection

Dataset	Network	Deepwalk	Node2vec	LINE	SDNE	CWCNE
SCN	SCN1	0.349	0.355	0.294	0.337	0.358
	SCN2	0.341	0.337	0.289	0.341	0.349
ACN	ACN1	0.423	0.429	0.342	0.425	0.425
	ACN2	0.411	0.423	0.318	0.412	0.415
FCN	FCN1	0.462	0.481	0.351	0.477	0.487
	FCN2	0.468	0.479	0.357	0.472	0.479
PCN	PCN1	0.440	0.460	0.403	0.429	0.463
	PCN2	0.439	0.452	0.398	0.424	0.457
WCN	WCN1	0.360	0.368	0.351	0.341	0.368
	WCN2	0.357	0.365	0.364	0.352	0.367
MCN	MCN1	0.516	0.503	0.486	0.529	0.572
	MCN2	0.504	0.491	0.481	0.523	0.548

表 2 展示了各类耦合网络节点向量在不同数据集上的模块度. 不同数据集上的值均为最大模块度对应的经验值. 在社交耦合网络、影视耦合网络、诗词耦合网络和人工耦合网络等 4 个数据集上, CWCNE 模型均获得了最佳的社团划分结果. 在学术耦合网络和著作耦合网络等 2 个数据集上, CWCNE 模型略逊于 node2vec 和 SDNE 方法.

表 3 展示了各类耦合网络节点向量在单网络上的社团划分结果. 在 SCN, FCN, PCN, WCN, MCN 等数据集上, CWCNE 模型的模块度均优于其他基准方法, 可见 CWCNE 方法生成的向量比其他仅用单网络信息生成节点向量保留了更多网络信息.

5.4　耦合网络主体识别

耦合网络主体识别是耦合网络分析的基础课题, 也是耦合网络节点表示十分重要的应用领域, 指在耦合网络中的多个网络之间, 识别出相同的主体, 即多个网络中具有对应关系的节点 [41]. 例如某个用户既使用 QQ 与好友进行交互, 又使用微信与好友进行交互, 那么在研究由 QQ 和微信构成耦合网络时, 该用户在两个网络中对应的节点即为相同主体. 通过耦合网络主体识别任务, 可以验证耦合网络节点向量是否包含了尽量多的耦合节点信息. 使用准确率作为评价指标, 准确率越高, 则其对应的耦合网络节点向量的质量越高. 在本实验中, 首先使用耦合网络节点向量计算节点间相似度, 然后对结果

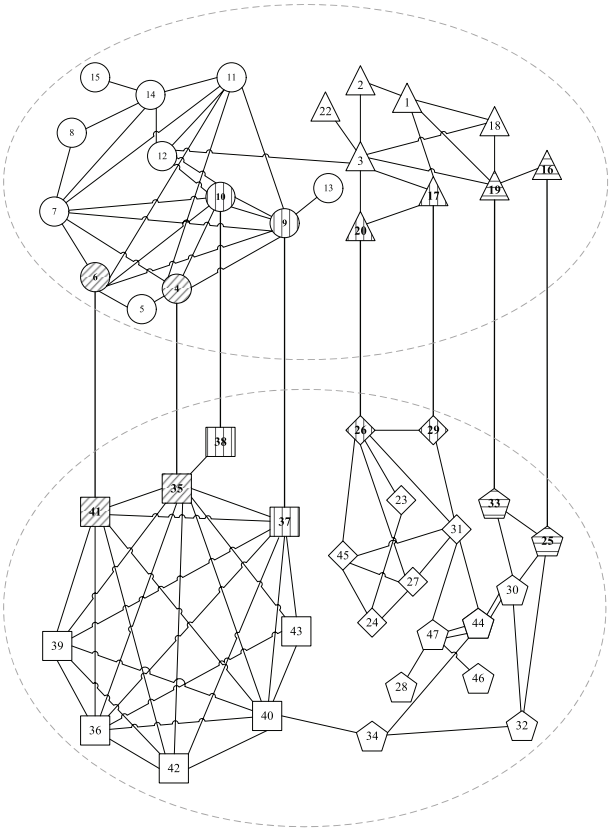

图 5 社交耦合网络数据集上主体识别结果

Figure 5 The recognition results of the main body of social coupled network

进行排序, 最后选取 Top5 作为潜在相同实体. 其中已知节点比例为 70%, 待识别节点比例为 30%.

图 5 为社交耦合网络数据集中, 耦合网络主体识别结果. 其中竖线填充节点为已知耦合节点, 斜线填充节点为被识别的耦合节点, 横线填充节点为未被识别的耦合节点. CWCNE 模型比较依赖网络结构信息, 从社交耦合网络的识别结果可以看出, 耦合节点间的邻边越多, 越易于识别出耦合节点, 如节点对 6, 41 和 4, 35 这些耦合节点在两个网络中分别都具有较多的邻边 (节点 6 与同侧耦合节点有 2 条邻边; 节点 4 与同侧耦合节点有 2 条邻边; 节点 41 与同侧耦合节点有 2 条邻边; 节点 35 与同侧耦合节点有 3 条邻边). 而未被识别的耦合节点, 如节点对 19, 33 和 16, 25 则存在较少的邻边 (节点 19 与同侧耦合节点有 1 条邻边; 节点 16 与同侧耦合节点有 1 条邻边; 节点 33 与同侧耦合节点有 1 条邻边; 节点 2 与同侧耦合节点有 1 条邻边), 所以不易识别出来.

表 4 为不同数据集下, 各方法在耦合网络主体识别任务中的准确率. 在 6 组数据集下, 由于每次随机游走所得的节点序列可能不同, 本实验在各算法参数设置不变的情况下, 进行十次实验并取结果平均值作为最终结果. 实验表明, CWCNE 模型均取得了最佳的识别效果. 其中在 PCN 下取得了最好的识别效果, 识别准确率达到 73.46%, 相较于其他网络, 诗词耦合网络 PCN 更偏向于文本网络, 所以更易识别.

图 6 展示了不同已知耦合节点比例对耦合节点识别准确率的影响. 分别设置已知耦合节点比例为 30%, 50%, 70%, 90%. 在不同数据集上, 随着已知耦合节点比例的增加, 耦合节点识别准确率也明显增加. 当已知耦合节点比例为 30% 时, 实体识别准确率很低, 基本没有识别能力. 当已知耦合节点比例

表 4　不同数据集耦合网络主体识别准确率

Table 4　The recognition accuracy of the main body of the coupled network

Dataset	Deepwalk	Node2vec	LINE	SDNE	CWCNE
SCN	0.187	0.187	0.174	0.192	0.250
ACN	0.389	0.391	0.382	0.391	0.394
FCN	0.437	0.437	0.425	0.439	0.452
PCN	0.692	0.687	0.673	0.698	0.735
WCN	0.517	0.526	0.527	0.531	0.564
MCN	0.604	0.618	0.607	0.611	0.635

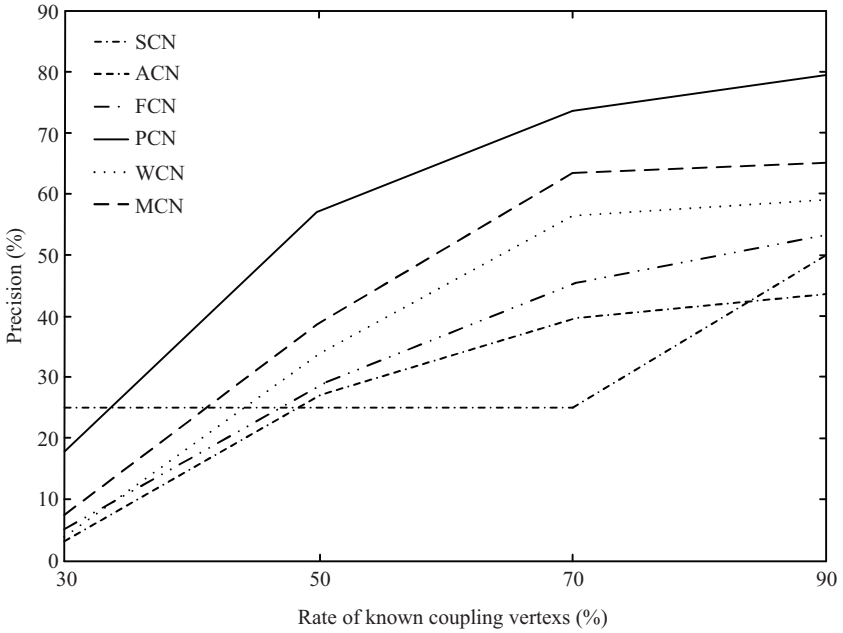

图 6　已知耦合节点比例对主体识别的影响

Figure 6　The influence of the known proportion of the coupling nodes on the body recognition

达到 90% 时, CWCNE 在不同数据集上的主体识别准确率均超过 0.4, 并在诗词耦合网络数据上取得了最佳的识别效果, 识别准确率达到 0.791.

由于主体识别是针对耦合网络的网络分析任务, 并能通过主体识别准确率评估耦合网络节点表示有效性, 因此本文对以上实验结果使用 Z 检验进行了显著性分析, 结果如表 5 所示. 其中, A vs. B 表示 A 算法比 B 算法准确率高, 标有 ** 的值表示 p 值小于 0.01, 说明算法性能差异显著.

5.5　标签分类

标签分类问题将节点属性作为标签, 通过标签分类问题, 可以验证耦合网络节点向量的分类能力, 使用准确率、召回率和值作为评价指标, 若值越高, 则其对应的耦合网络节点向量的分类能力越强. 在本实验中, 均使用 SVM 作为分类器. 其中训练集比例 70%, 测试集比例 30%.

在社交耦合网络中进行标签分类任务. 使用用户性别进行二分类任务, 使用用户分组进行多分类 (9 类) 任务.

表 5 耦合网络主体识别结果显著性分析

Table 5 Significant analysis of subject recognition results

Dataset (%)	SCN	ACN	FCN	FCN	WCN	MCN
CWCNE vs. DeepWalk	6.3**	0.5	1.5**	4.3**	4.7**	3.1**
CWCNE vs. Node2vec	6.3**	0.3	1.5**	4.8**	3.8**	1.7**
CWCNE vs. LINE	7.6**	1.2**	2.7**	6.2**	3.7**	2.8**
CWCNE vs. SDNE	5.8**	0.3**	1.3**	3.7**	3.3 **	2.4**

表 6 不同数据集二标签分类结果

Table 6 Two label classification results

Dataset	Measure	Deepwalk	Node2vec	LINE	SDNE	CWCNE
SCN	P	0.751	0.872	0.793	0.852	0.875
	R	0.681	0.603	0.640	0.623	0.619
	$F1$	0.714	0.712	0.708	0.720	0.725
ACN	P	0.814	0.836	0.803	0.837	0.851
	R	0.659	0.729	0.631	0697	0.783
	$F1$	0.728	0.778	0.707	0.761	0.815
FCN	P	0.688	0.687	0.675	0.683	0.693
	R	0.573	0.598	0.551	0.579	0.604
	$F1$	0.625	0.629	0.607	0.628	0.645
PCN	P	0.937	0.922	0.908	0.925	0.950
	R	0.892	0.872	0.883	0.868	0.895
	$F1$	0.913	0.896	0.895	0.896	0.921

在学术耦合网络中进行标签分类任务. 使用用户性别进行二分类任务, 使用研究领域进行多分类 (6 类) 任务.

在影视耦合网络中进行标签分类任务. 使用影人性别进行二分类任务, 使用影人星座进行多分类 (12 类) 任务.

在诗词耦合网络中进行标签分类任务. 使用音律平仄进行二分类任务.

在著作耦合网络中进行标签分类任务. 使用研究领域进行多分类 (8 类) 任务.

表 6 展示了不同模型在 2 标签分类任务中的结果. CWCNE 模型在各个数据集上的预测效果均略高于其他基准方法, 可以较准确地预测出标签属性. 在诗词耦合网络中, 由于音律平仄较为规范, 所以准确率较高, 各方法的准确率均在 0.90 以上.

图 7 为二标签分类任务的曲线. 在 PCN 数据集上分类效果最好, 曲线下面积达到 0.79. 在 SCN 和 CAN 数据集上的分类效果基本持平, 曲线下面积分别为 0.60 和 0.58. 在 FCN 数据集上分类效果最差, 曲线下面积仅有 0.39.

表 7 展示了不同模型在多标签分类任务中的结果对比. CNE 模型在 SCN, WCN 等数据集上略高于基准方法, 其中 ACN 数据集上的值最高, 达到了 0.363, PCN 上的值最低, 仅为 0.128. 类别数量对 CWCNE 模型的分类能力影响很大, 类别数量越少, 分类效果越好.

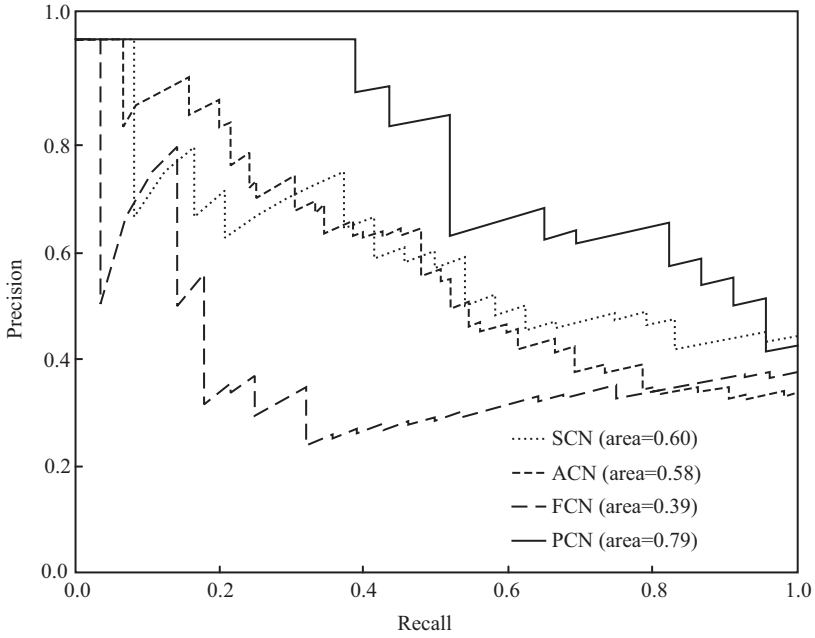

图 7　二标签分类 P-R 曲线

Figure 7　Two label classification P-R curve

表 7　多分类任务准确率对比

Table 7　Accuracy rate of multi classification task

Dataset	Measure	Deepwalk	Node2vec	LINE	SDNE	CWCNE
SCN ($N=9$)	P	0.276	0.301	0.283	0.298	0.307
	R	0.201	0.248	0.237	0.246	0.246
	$F1$	0.232	0.272	0.258	0.270	0.273
ACN ($N=6$)	P	0.338	0.374	0.352	0.369	0.375
	R	0.308	0.352	0.318	0.341	0.352
	$F1$	0.322	0.363	0.334	0.354	0.363
PCN ($N=12$)	P	0.122	0.173	0.143	0.164	0.183
	R	0.073	0.102	0.088	0.097	0.097
	$F1$	0.091	0.128	0.109	0.122	0.128
WCN ($N=8$)	P	0.325	0.321	0.315	0.322	0.336
	R	0.179	0.183	0.168	0.173	0.185
	$F1$	0.231	0.233	0.219	0.225	0.238

5.6　标签分类参数分析

标签分类问题将节点属性作为标签, 通过标签分类问题, 可以验证耦合网络节点向量的分类能力, 使用准确率作为评价指标, 准确率越高, 则其对应的耦合网络节点向量的分类能力越强. 在本实验中, 均使用 SVM 作为分类器. 其中训练集比例 70%, 测试集比例 30%.

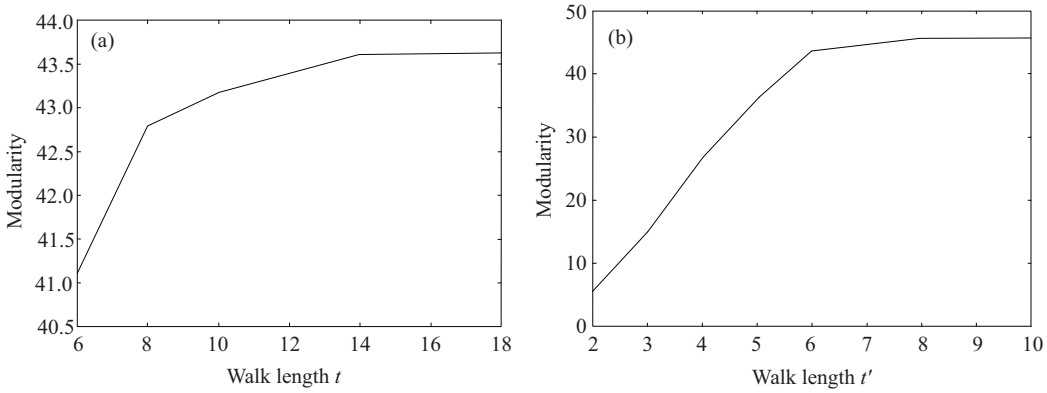

图 8 步长对耦合节点向量的影响

Figure 8 The effect of step size on the coupling node vector. (a) Current network step size; (b) coupling network step size

5.6.1 游走步长对结果的影响

游走步长是 CWCNE 模型中的重要参数, 对步长 t, t' 进行分析, 可以获取到其最佳经验值, 在实际应用中有意义. 本实验使用社团划分作为测试任务, 使用模块度作为评价指标, 模块度最大值对应的步长 t, t', 即为最佳经验值. 在影视耦合网络数据集上对游走步长进行分析. 图 8(a) 说明, 在一定范围内, 随着步长 t 的增大, 社团划分的模块度小幅提升, 模型测试效果较为稳定, 当步长 t 达到 14 时, 社团划分的模块度基本不再上升. 图 8(b) 说明, 随着步长 t' 的增大, 社团划分的模块度提升明显, 当步长 t' 达到 6 时, 社团划分的模块度趋于稳定.

5.6.2 向量维度对结果的影响

耦合网络向量维度 d 也是 CWCNE 模型中的一个重要参数, 在保证耦合网络节点向量质量的前提下, 尽量降低向量维度 d 可以减少存储空间和计算量. 本实验使用实体识别作为测试任务, 使用准确率作为评价指标. 期望在准确率达到预期值的情况下, 找到最小的向量维度 d. 在影视耦合网络数据集上对向量维度进行分析.

图 9 显示, 随着向量维度的增高, 社团划分的模块度增高, 但提升并不明显, 当向量维度达到 50 维时, 实体识别的结果趋于稳定. 说明在影视耦合网络数据下, 50 维的耦合网络节点向量即可保存足够的耦合网络节点信息.

5.6.3 上下文窗口对结果的影响

在耦合约束随机游走阶段, 上下文窗口 ω 也是一个重要的参数. 直观来看, ω 的值越大, 其对应的耦合网络节点向量的表征能力应该越强. 但在实验过程中, ω 的值对实验结果并无明显影响. 其原因在于, 节点序列的生成共包含两个阶段, 在耦合约束随机游走阶段, 原始序列已经包含了随机性, 在使用上下文窗口进行采样的阶段, 则 ω 参数的影响较小. 在完整的节点序列生成过程中, t, t' 及参数为主要参数, 而 ω 参数的影响则几乎可以忽略.

6 总结与未来工作

本文针对耦合网络提出了耦合网络嵌入模型 CWCNE. 针对耦合网络的特性, 改进了嵌入方法中

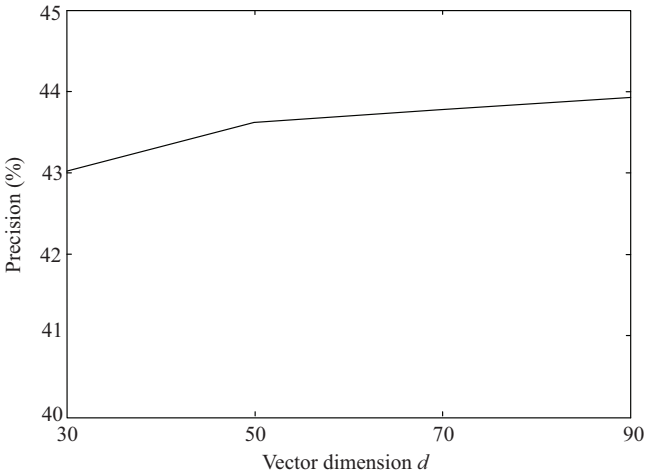

图 9　向量维度对耦合节点向量的影响

Figure 9　The effect of vector dimension on the coupling node vector

的游走算法, 提出了一种网络间的随机游走策略; 同时改进了模型的训练方法, 使用网络间迭代训练的方式来学习模型参数. 在社交耦合网络、学术耦合网络、影视耦合网络、诗词耦合网络、著作耦合网络等 5 组数据集上验证了 CWCNE 的有效性. 并在社团划分、实体识别、标签预测等任务上取得了良好的结果. 在耦合网络中, 如何将不同网络中的属性信息引入耦合网络节点向量是一个值得研究的方向. 另外将耦合网络间的用户动态交互信息引入耦合网络节点向量也是值得研究的方向.

参考文献

1 Tan S L, Guan Z Y, Cai D, et al. Mapping users across networks by manifold alignment on hypergraph. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, Québec City, 2014. 159–165

2 Zhao D W, Wang L H, Li S D, et al. Immunization of epidemics in multiplex networks. Plos One, 2014, 9: e112018

3 Wang W, Tang M, Zhang H F, et al. Epidemic spreading on complex networks with general degree and weight distributions. Phys Rev E, 2014, 90: 042803

4 Zhang H F, Shu P P, Tang M, et al. Preferential imitation of vaccinating behavior can invalidate the targeted subsidy on complex network. 2015. ArXiv: 1503.08048

5 Lerman K, Ghosh R. Information contagion: an empirical study of the spread of news on digg and twitter social networks. Comput Sci, 2010, 52: 166–176

6 Li R Q, Tang M, Hui P M. Epidemic spreading on multi-relational networks. J Phys, 2013, 62: 504–510

7 Cozzo E, Baños R A, Meloni S, et al. Contact-based social contagion in multiplex networks. Phys Rev E, 2013, 88: 050801

8 Zhang X. Multilayer networks science: concepts, theories and data. Complex Syst Complex Sci, 2015, 12: 103–107 [张欣. 多层复杂网络理论研究进展: 概念, 理论和数据. 复杂系统与复杂性科学, 2015, 12: 103–107]

9 Mollgaard A, Zettler I, Dammeyer J, et al. Measure of node similarity in multilayer networks. 2016. ArXiv: 1606.00715

10 Granell C, Gómez S, Arenas A. Competing spreading processes on multiplex networks: awareness and epidemics. Phys Rev E, 2014, 90: 012808

11 Zafarani R, Liu H. Connecting users across social media sites: a behavioral-modeling approach. In: Proceedings of the 19th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Chicago, 2013. 41–49

12 Zafarani R, Liu H. Connecting corresponding identities across communities. In: Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media, San Jose, 2009. 354–357

13 Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality. In: Proceedings of the 27th Conference on Neural Information Processing Systems, Lake Tahoe, 2013. 3111–3119

14 Matsuno R, Murata T. MELL: effective embedding method for multiplex networks. In: Proceedings of the 27th World Wide Web Conference, Lyon, 2018. 1261–1268

15 Le Q V, Mikolov T. Distributed representations of sentences and documents. Comput Sci, 2014, 4: 1188–1196

16 Perozzi B, Alrfou 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

17 Gligorijevic V, Panagakis Y, Zafeiriou S. Non-negative matrix factorizations for multiplex network analysis. IEEE Trans Pattern Anal, 2018, 41: 928–940

18 Lee D D, Seung H S. Learning the parts of objects by non-negative matrix factorization. Nature, 1999, 401: 788–791

19 Feng T, Li S Z, Shum H Y, et al. Local non-negative matrix factorization as a visual representation. In: Proceedings of the 2nd International Conference on Development and Learning, Cambridge, 2002. 178–183

20 Guillamet D, Vitriá J, Schiele B. Introducing a weighted non-negative matrix factorization for image classification. Pattern Recogn Lett, 2003, 24: 2447–2454

21 Wang Y, Jia Y D, Hu C B, et al. Fisher non-negative matrix factorization for learning local features. In: Proceedings of the Asian Conference on Computer Vision, Jeju Island, 2004. 27–30

22 Gao Y, Church G. Improving molecular cancer class discovery through sparse non-negative matrix factorization. Bioinformatics, 2005, 21: 3970–3975

23 Yi L, Rong J, Liu Y. Semi-supervised multi-label learning by constrained non-negative matrix factorization. In: Proceedings of the 21st National Conference on Artificial Intelligence, Boston, 2006. 421–426

24 Carmonasaez P, Pascualmarqui R D, Tirado F, et al. Biclustering of gene expression data by non-smooth non-negative matrix factorization. BMC Bioinform, 2006, 7: 205–208

25 Tang J, Qu M, Wang M Z, et al. LINE: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, Florence, 2015. 1067–1077

26 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

27 Zhou N, Zhao W X, Zhang X, et al. A general multi-context embedding model for mining human trajectory data. IEEE Trans Knowl Data Eng, 2016, 28: 1945–1958

28 Cui P, Wang X, Pei J, et al. A survey on network embedding. IEEE Trans Knowl Data Eng, 2019, 31: 833–852

29 Wang D X, Cui P, Zhu W 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

30 Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks. 2016. ArXiv: 1609.02907

31 Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs. In: Proceedings of the 31st Conference on Neural Information Processing Systems, Long Beach, 2017. 1024–1034

32 Chen J, Ma T F, Xiao C. Fastgcn: fast learning with graph convolutional networks via importance sampling. 2018. ArXiv: 1801.10247

33 Pan S R, Hu R Q, Long G D, et al. Adversarially regularized graph autoencoder for graph embedding. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, 2018. 2609–2615

34 Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. Science, 2006, 313: 504–507

35 Wang H W, Wang J, Wang J L, et al. GraphGAN: graph representation learning with generative adversarial nets. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, New Orleans, 2018. 2508–2515

36 Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative adversarial nets. In: Proceedings of the 28th Conference on Neural Information Processing Systems, Montreal, 2014. 2672–2680

37 Qu M, Tang J, Shang J B, et al. An attention-based collaboration framework for multi-view network representation learning. In: Proceedings of the 26th ACM International Conference on Information and Knowledge Management, Singapore, 2017. 1767–1776

38 Xu L C, Wei X K, Cao J N, et al. Embedding of embedding (eoe): joint embedding for coupled heterogeneous networks. In: Proceedings of the 10th ACM International Conference on Web Search and Data Mining, Cambridge, 2017. 741–749

39 Tang J, Cai K K, Su Z, et al. BigNet 2016: first workshop on big network analytics. In: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, München, 2016. 2505–2506

40 Fortunato S. Community detection in graphs. Phys Rep, 2010, 486: 75–174
41 Kong X, Zhang J, Yu P S. Inferring anchor links across multiple heterogeneous social networks. In: Proceedings of the 22nd ACM International on Conference on Information and Knowledge Management, New York, 2013. 179–188

Coupling network vertex representation learning based on network embedding method

Zhongming HAN[1,2], Dan LIU[1], Chenye ZHENG[1], Wen LIU[1], Dagao DUAN[1*] & Jian DONG[3]

1. *College of Computer and Information Engineering, Beijing Technology and Business University, Beijing* 100048*, China*;
2. *Beijing Key Laboratory of Food Safety and Big Data Technology, Beijing* 100048*, China*;
3. *Key Laboratory of Information Network Security Ministry of Public Security, the Third Research Institute of the Ministry of Public Security, Shanghai* 200031*, China*
* Corresponding author. E-mail: duandg@th.btbu.edu.cn

Abstract Network representation learning is a basic problem in network data analysis. By learning network representation vectors, network vertices can be represented more accurately. With the development of deep learning, embedding methods have been widely used for network vertex representation learning. Providing that network data have changed in terms of their scale and modality, the research focus gradually shifted from single network mining to coupling network mining. This paper first analyzes the research status of embedding methods for single networks and then compares their advantages and disadvantages. Furthermore, the paper presents a model called CWCNE for coupling network embedding. The random walk and training algorithms of the proposed model are improved to adapt to coupling network features. The validity of the proposed model was verified using social, academic, film, poetry, and work coupling network data. Good results were obtained on community detection, entity recognition, and label classification tasks.

Keywords network embedding, vertex vector, coupling network, representation learning, community detection, entity recognition, label classification

Zhongming HAN was born in 1972. He received his Ph.D. degree from Donghua University, Shanghai, in 2006. Currently, he is a professor at Beijing Technology and Business University. His research interests include big data, web data mining, and social computing.

Dan LIU was born in 1995. He is currently working toward his master's degree in computer science at Beijing Technology and Business University. His research interests include machine learning and Internet data mining.

Dagao DUAN was born in 1976. He received his Ph.D. degree from Beijing University of Posts and Telecommunications, Beijing, in 2005. Currently, he is an associate professor at Beijing Technology and Business University. His research interests include social computing and multimedia information processing.

Jian DONG was born in 1974. He received his Ph.D. degree from the People's Public Security University of China, Beijing, in 2009. Currently, he is a senior researcher at the Third Research Institute of the Ministry of Public Security. His research interests include cybercrime investigation and digital evidence.