



计算机科学与探索

Journal of Frontiers of Computer Science and Technology

ISSN 1673-9418, CN 11-5602/TP

《计算机科学与探索》网络首发论文

题目：图嵌入模型综述
作者：袁立宁，李欣，王晓冬，刘钊
网络首发日期：2021-09-13
引用格式：袁立宁，李欣，王晓冬，刘钊. 图嵌入模型综述. 计算机科学与探索.
<https://kns.cnki.net/kcms/detail/11.5602.TP.20210913.1016.002.html>



网络首发：在编辑部工作流程中，稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定，且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式（包括网络呈现版式）排版后的稿件，可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定；学术研究成果具有创新性、科学性和先进性，符合编辑部对刊文的录用要求，不存在学术不端行为及其他侵权行为；稿件内容应基本符合国家有关书刊编辑、出版的技术标准，正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性，录用定稿一经发布，不得修改论文题目、作者、机构名称和学术内容，只可基于编辑规范进行少量文字的修改。

出版确认：纸质期刊编辑部通过与《中国学术期刊（光盘版）》电子杂志社有限公司签约，在《中国学术期刊（网络版）》出版传播平台上创办与纸质期刊内容一致的网络版，以单篇或整期出版形式，在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊（网络版）》是国家新闻出版广电总局批准的网络连续型出版物（ISSN 2096-4188，CN 11-6037/Z），所以签约期刊的网络版上网络首发论文视为正式出版。

图嵌入模型综述

袁立宁¹, 李欣², 王晓冬³, 刘钊⁴⁺

1. 中国人民公安大学 信息网络安全学院, 北京 100038
 2. 天津市公安局河西分局 科技信息化支队, 天津 300202
 3. 天津市公安局河东分局 科技信息化支队, 天津 300171
 4. 中国人民公安大学 新型犯罪研究中心, 北京 100038
- + 通信作者 E-mail: liuzhao@ppsuc.edu.cn

摘要: 图分析用于深入挖掘图数据的内在特征, 然而图作为非欧几里得数据, 传统的数据分析方法普遍存在较高的计算量和空间开销。图嵌入是一种解决图分析问题的有效方法, 其将原始图数据转换到低维空间并保留关键信息, 从而提升节点分类、链接预测、节点聚类下游任务的性能。与以往的研究不同, 本文同时对静态图和动态图嵌入文献进行全面回顾, 提出一种全新的静态图嵌入和动态图嵌入通用分类方法, 即基于矩阵分解的图嵌入、基于随机游走的图嵌入、基于自编码器的图嵌入、基于图神经网络的图嵌入和基于其他方法的图嵌入。其次, 对静态图和动态图方法的理论相关性进行分析, 对模型核心策略、下游任务和数据集进行全面总结。最后, 提出了四个图嵌入的潜在研究方向。

关键词: 图嵌入; 静态图嵌入; 动态图嵌入; 随机游走; 图神经网络

文献标志码: A **中图分类号:** TP391

Graph Embedding Models: A Survey

YUAN Lining¹, LI Xin², WANG Xiaodong³, LIU Zhao⁴⁺

1. School of Information Network Security, People's Public Security University of China, Beijing 100038, China
2. Science and Informatization Division, Hexi Branch of Tianjin Public Security Bureau, Tianjin 300202, China
3. Science and Informatization Division, Hedong Branch of Tianjin Public Security Bureau, Tianjin 300171, China
4. Research Center for New Crimes, People's Public Security University of China, Beijing 100038, China

Abstract: Effective graph analysis methods can reveal the intrinsic characteristics of graph data. However, graph is non-Euclidean data, which leads to the high computation and space cost while applying traditional methods. Graph embedding is an efficient method for graph analysis. It converts original graph data into a low-dimensional space and retains key information to improve the performance of downstream tasks such as node classification, link prediction, and node clustering. Different from previous studies, our review focuses on both static and dynamic graph

基金项目: 中央高校基本科研业务费专项资金 (2019JKF425); 国家重点研发计划 (2018YFC0809800)。

This work was supported by the Fundamental Research Funds for the Central Universities (2019JKF425), and the National Key Research and Development Program of China (2018YFC0809800).

embedding. Firstly, we propose a universal taxonomy of static and dynamic methods for the first time, including matrix factorization based methods, random walk based methods, autoencoder based methods, graph neural networks and other embedding methods. Secondly, we analyze the theoretical relevance of static and dynamic methods, and comprehensively summarizes the core strategy, datasets and downstream tasks. Finally, we propose four potential research directions.

Key words: graph embedding; static graph embedding; dynamic graph embedding; random walk; graph neural network

图是复杂系统中常用的信息载体,可以表示现实中许多复杂关系,如社交网络^[1]、犯罪网络^[2]、交通网络^[3]等。图结构作为一种非欧几里得数据,很难直接应用卷积神经网络(Convolutional Neural Network, CNN)^[4]和循环神经网络(Recurrent Neural Network, RNN)^[5]等深度学习方法^[6]。为了构造用于图数据挖掘的特征表示,图嵌入将节点映射到低维空间,生成保留原始图中某些重要信息的低维向量。目前,图嵌入不仅在节点分类^[7]、链接预测^[8]、节点聚类^[9]、可视化^[10]等复杂网络上的机器学习任务中获得成功,还广泛用于社交影响力建模^[11]、内容推荐^[12]等现实任务。

早期的图嵌入算法主要用于数据降维,通过邻域关系构建相似度图,将节点嵌入低维向量空间,并保持相连节点向量的相似性。这类方法通常时间复杂度高,很难扩展到大型图上。近年来,图嵌入算法转向扩展性强的方法。例如,矩阵分解方法^[13]使用邻接矩阵的近似分解作为嵌入;随机游走法^[14]将游走序列输入到 Skip-Gram^[15]生成嵌入。这些方法利用图的稀疏性降低了时间复杂度。当前,很多综述^[16-21]对图嵌入方法进行了归纳与总结,但存在两大局限:一是部分综述仅涉及传统方法介绍,许多新模型没有纳入研究;二是这些综述只关注静态图嵌入或动态图嵌入,忽略了二者之间的关联性。

本文对图嵌入方法进行全面系统性综述,有以

下三个方面的贡献:(1)本文提出一种新的图嵌入分类法,同时对静态图和动态图方法进行分类;

(2)本文对现有模型进行系统性分析,为理解现有方法提供新视角;(3)本文提出了四个图嵌入的潜在研究方向。

论文其余部分安排如下。第一节引入图嵌入相关定义,列出文中使用的符号及含义。第二节提出图嵌入的分类方法,介绍每类中代表性模型,讨论静态模型和动态模型的相互关系。第三节介绍图嵌入的数据集及应用。第四节提出图嵌入的潜在研究方向。第五节全文内容进行总结。

1 图嵌入相关定义及符号表示

1.1 问题定义

定义 1 (图) 图通常表示为 $G=(V, E)$, 其中 V 表示节点集, E 表示边集。

定义 2 (静态图) 给定图 $G=(V, E)$, 如果节点和边的状态不随时间变化, 图 G 为静态图。

定义 3 (动态图) 动态图可以按时间分成一系列演化图 $\mathcal{G}=(G^1, G^2, \dots, G^T)$, T 表示演化图的数量。每个演化图 $G^t=(V^t, E^t)$ 表示节点和边在 t 时刻的状态。动态图包含快照型和连续时间型(见图 1)。快照型动态图按时间序列将动态图分解为等间隔的静态图;连续时间型用多个时间戳标记每条边来保留节点间的连接变化。

定义 4（一阶相似度） 一阶相似度描述节点之间的成对邻近度。如果节点 v_i 和 v_j 之间存在一条边， v_i 和 v_j 的一阶相似度为边权重 w_{ij} ；如果在 v_i 和 v_j 之间没有边，一阶相似度为 0。

定义 5（二阶相似度） 二阶相似度描述节点邻域结构的相似度。假设 $w_i = [w_{i1}, \dots, w_{im}]$ 表示节点 v_i 与其他节点的一阶相似度，那么 v_i 和 v_j 的二阶相似度由 w_i 和 w_j 的相似程度决定。

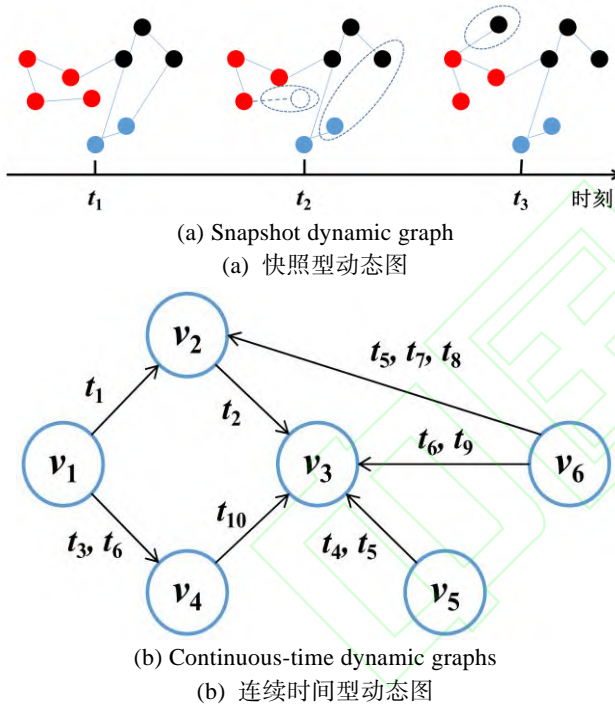


Fig.1 Snapshot and continuous-time dynamic graphs
图 1 快照型和连续时间型动态图

定义 6（图嵌入） 图嵌入将每个节点映射成低维向量表示（见图 2），同时保留了原始图中某些关键信息。映射函数通常定义为 $f: v_i \rightarrow Y_i \in \mathbb{R}^d$ ，其中 d 为嵌入向量的维度。

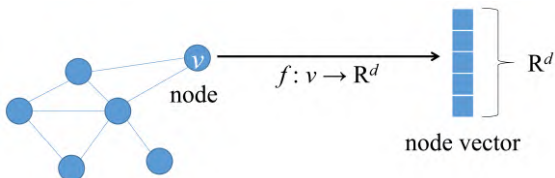


Fig.2 A general framework for graph embedding
图 2 图嵌入的一般过程

1.2 符号表示

本文常用符号及其定义见表 1。

Table 1 Symbols and definitions

表 1 符号及定义

符号	说明
A	邻接矩阵
D	顶点度矩阵
L	拉普拉斯矩阵
X	特征矩阵
S	相似度矩阵
W	系数矩阵
G	静态图
\mathcal{G}	动态图
V	节点集
E	边集
d	嵌入维度
Y	嵌入表示

2 图嵌入分类

本节提出一种新的分类方法，用于现有图嵌入模型的分类。按照模型所使用的算法原理将静态图和动态图模型同时分为五个大类：基于矩阵分解的图嵌入、基于随机游走的图嵌入、基于自编码器^[22]的图嵌入、基于图神经网络^[23]的图嵌入和基于其他方法的图嵌入。图 3 为分类思路及内容体系构建，图 4 为图嵌入模型的分类汇总。

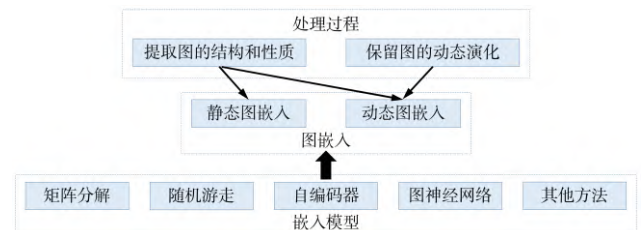


Fig.3 The content system of graph embedding
图 3 图嵌入内容体系

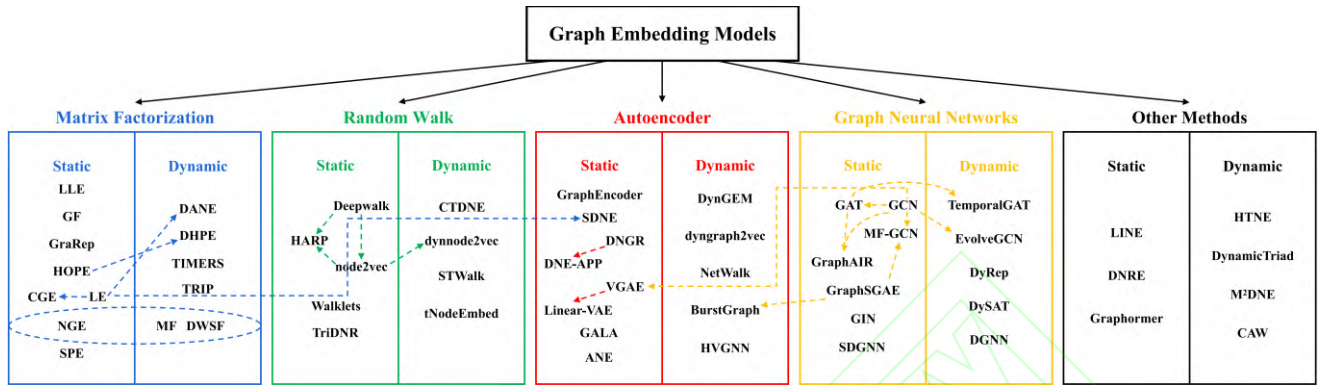


Fig.4 The categorization and summary of graph embedding models

图 4 图嵌入模型分类汇总

2.1 基于矩阵分解的图嵌入

基于矩阵分解的图嵌入通过分解节点关系矩阵获得低维嵌入。不同的关系矩阵采用的分解方法不同,例如邻接矩阵通常使用奇异值分解(Singular Value Decomposition, SVD)的方法生成节点嵌入,而属性矩阵通常使用特征值分解的方法。

2.1.1 基于矩阵分解的静态图嵌入

基于矩阵分解的静态图嵌入模型对节点关联信息矩阵和属性信息矩阵进行特征分解,然后将分解得到的属性嵌入和结构嵌入进行融合,生成节点的低维嵌入表示。图 5 说明了矩阵分解和静态图嵌入生成的一般过程。

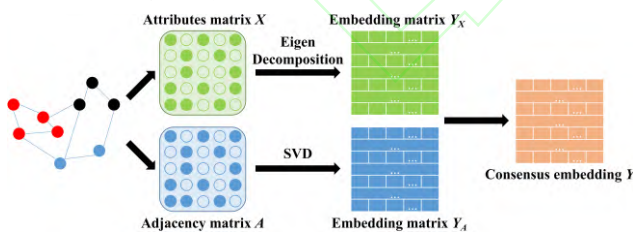


Fig.5 The framework of eigenvalue factorization in static graph embedding

图 5 基于矩阵分解的静态图嵌入模型框架

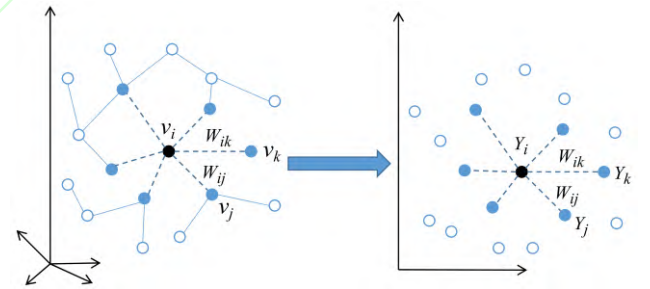
局部线性映射 (Locally Linear Embedding, LLE)^[24]将每个节点表示为相邻节点的线性组合,构造邻域保持映射(见图 6)。具体实现分为三步:

(1) 以某种度量方式(如欧氏距离)选择 k 个邻近节点;(2) 由 k 个近邻线性加权重构节点,并最小化节点重建误差获得最优权重;(3) 最小化最优

权重构建的目标函数生成 Y 。目标函数的表达式为:

$$L = \sum_i \left\| Y_i - \sum_j W_{ij} Y_j \right\|^2 \quad (1)$$

式中, W_{ij} 为节点 i 与节点 j 的权重系数。作为一种重要的降维算法, LLE 在降维时能够保持样本的局部特征,并通过稀疏矩阵特征分解的方式适当降低了计算复杂度。但是,该模型对 k 值的选择十分敏感,对最终的降维结果有极大影响。

Fig.6 Steps of LLE^[24]图 6 LLE 步骤^[24]

(1) Reconstruct node with linear weights; (2) Map to embedded coordinates

(1) 使用线性权重重构节点;(2) 将节点映射到嵌入维度

许多图分解和重构的目标函数都使用节点向量来确定边的建立。GF(Graph Factorization)^[13]假设边的存在信息可以由节点向量内积 $\langle Y_i, Y_j \rangle$ 有效地捕捉。为了获得嵌入, GF 对邻接矩阵进行分解,并利用均值方差和 L2 正则项构建目标函数:

$$L = \frac{1}{2} \sum_{(i,j) \in E} (A_{ij} - \langle Y_i, Y_j \rangle)^2 + \frac{\lambda}{2} \sum_i \|Y_i\|^2 \quad (2)$$

由上式特性可知，在节点数量较多、嵌入维度较大时，模型计算会产生极大的内存占用。为此，GF设计了一种最小化簇外一阶邻居数量的图分割算法，保证图结构信息无损，提升模型计算效率。

GraRep^[25]分别构建 1 到 k 步的对数转移概率矩阵 $\{T^1, T^2, \dots, T^k\}$ ，将 T^k 中所有负值元素替换为 0，使 T^k 为正 k 步对数转移概率矩阵，以减少噪声^[26]。最后，使用 SVD 分解 T^k 得到节点表示 Y^k ，并将 $\{Y^1, Y^2, \dots, Y^k\}$ 进行合并，得到最终嵌入 Y 。GraRep 能够在嵌入中整合全局结构信息，但训练过程中涉及矩阵运算和 SVD，计算复杂度极高，难以扩展到大规模图数据。

非对称传递性可以刻画有向边之间的关联，有助于捕捉图的结构。为了保留有向图中的非对称传递性，HOPE^[27]采用了一种保持高阶相似度的嵌入方法，在保留非对称传递性的向量空间中生成图嵌入（见图 7），训练中使用 L2 范数进行优化：

$$L = \|S - Y_s Y_t^T\|_F^2 \quad (3)$$

式中， Y_s 为源嵌入和 Y_t 为目标嵌入。许多相似性度量可以反映非对称传递性，如 Katz 指标^[28]、Adamic-Adar 分数等，用于构建相似度矩阵 S 。此外，HOPE 使用广义 SVD (Generalized Singular Value Decomposition, GSVD)^[29] 分解 S ，适当降低了计算复杂度，但是低阶 GSVD 逼近能力有限，限制了模型表达能力。

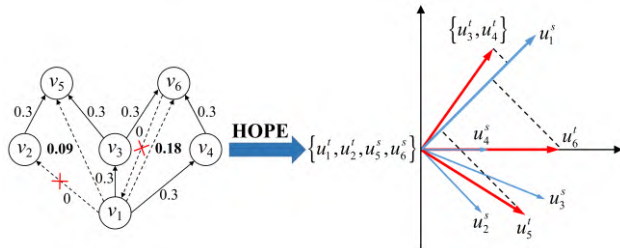


Fig.7 Framework of HOPE^[27]. The left is a input graph, and the right is the embedding space. In graph, the solid arrows represent observed edges and the numbers along with the solid arrows are edge weights. The numbers along with the dashed arrows is Katz proximity

图 7 HOPE 模型框架^[27]：左侧是输入有向图，右侧是嵌入空间。实心箭头表示观察到的有向边和边的权重。虚线箭头表示 Katz 相似度的值

在原始图上，本征图用于保留节点间有利关联，惩罚图用于保留节点间的不利关联。为了综合本征图和惩罚图的特点，NGE (Non-Negative Graph Embedding)^[30]引入非负矩阵分解 (Non-negative Matrix Factorization, NMF)^[31]生成嵌入表示。NMF 将数据矩阵分解成低阶非负基矩阵 W 和非负系数矩阵 Y ，并使用 L1 损失作为目标函数。在此基础上，NGE 将 Y 和 W 分成两部分： $Y = [Y^1, Y^2]$ ； $W = [W^1, W^2]$ 。由于 (W_1, Y_1) 和 (W_2, Y_2) 是互补空间，可以通过叠加的方式重构原始数据，所以 Y^2 的目标函数可以由惩罚图目标函数进行转换。将 NMF 的 L1 损失和 Y^1 和 Y^2 目标函数相结合，可得 NGE 的目标函数为：

$$L = \sum_{i,j} \|y_i^1 - y_j^1\| A_{ij} + \sum_{i,j} \|y_i^2 - y_j^2\| A_{ij}^p + \lambda \|Y - WY\| \quad (4)$$

式中， $Y^1 = [y_1^1, y_2^1, \dots, y_N^1]$ 表示生成的低维嵌入， $Y^2 = [y_1^2, y_2^2, \dots, y_N^2]$ ， λ 表示平衡图嵌入和数据重构的正值参数。NGE 提出了非负图嵌入的通用公式，并且同时使用征图和惩罚图能够有效重构原始数据，但是该模型的计算成本相对较高，不能在保持非负性更新规则的同时显式控制基矩阵的稀疏性。

拉普拉斯特征映射 (Laplacian Eigenmaps, LE)^[32]与 LLE 相似，也是从局部近似的角度构建数据之间的关系。具体而言，LE 使用邻接矩阵建立包含局部结构信息的嵌入表示，并要求相连节点在嵌入空间中尽可能地靠近。因此，LE 的目标函数为：

$$L = \sum_{i,j} A_{ij} \|y_i - y_j\|^2 \quad (5)$$

由上式可知，LE 的目标函数强调权值大的节点对，弱化权值小的节点对，导致原始图中的局部拓扑结构被破坏。为了增强图嵌入的局部拓扑保持性，柯西图嵌入 (Cauchy Graph Embedding, CGE)^[33]引入距离的反函数来保持节点在嵌入空间中的相似关系。因此，CGE 将 LE 的目标函数改写为：

$$L = \sum_{i,j} \frac{A_{ij}}{|y_i - y_j|^2 + \sigma^2} \quad (6)$$

相比 LE，CGE 的目标函数更加强调整短距离，即确保局部越邻近的节点在嵌入空间的距离越接近。

为了保持图的全局拓扑结构, SPE(Structure Preserving Embedding)^[34]通过一组线性不等式学习包含节点连接关系的低阶核矩阵获得嵌入表示。SPE 以核矩阵 \mathbf{K} 为输入, 使用连接算法 f 输出关系矩阵 $\tilde{\mathbf{A}} = f(\mathbf{K})$ 。通过比较邻接矩阵 \mathbf{A} 与关系矩阵 $\tilde{\mathbf{A}}$ 的差值可以评估 \mathbf{K} 产生的嵌入对图结构信息的保留程度。与 LE 相比, SPE 在图的可视化以及无损压缩方面有显著提升, 证明了在降维算法中引入结构保持约束可以更加准确地表示高维数据。

2.1.2 基于矩阵分解的动态图嵌入

从矩阵分解的角度看, 图的动态演化实质上是原始矩阵的不断变化。基于矩阵分解的动态图方法利用特征分解构造图的高阶相似度矩阵, 然后利用矩阵摄动理论^[35]更新图的动态信息。矩阵摄动理论可以高效更新图的高级特征对, 同时避免了每个时刻的重新计算嵌入矩阵。图 8 是基于矩阵分解的动态图嵌入的一般过程。

DANE^[36]采用分布式框架(见图 9): 离线部分, 采用最大化相关性^[37]的方法, 引入 $\mathbf{p}_t \mathbf{A}$ 和 $\mathbf{p}_t \mathbf{X}$

使拓扑嵌入 $\mathbf{Y}_t \mathbf{A}$ 和特征嵌入 $\mathbf{Y}_t \mathbf{X}$ 投影后的相关性最大化, 捕捉图结构和节点属性的依赖关系, 然后将前 d 个特征向量进行拼接, 保持 $\mathbf{Y}_t \mathbf{A}$ 和 $\mathbf{Y}_t \mathbf{X}$ 在表示上一致; 在线部分, 使用矩阵摄动理论更新嵌入的特征值和特征向量。DANE 有效捕捉了拓扑结构和节点属性的相关性, 保留了图的动态信息, 但是模型的在线部分仅使用一阶矩阵摄动理论更新嵌入表示, 并未考虑图的高阶相似性。

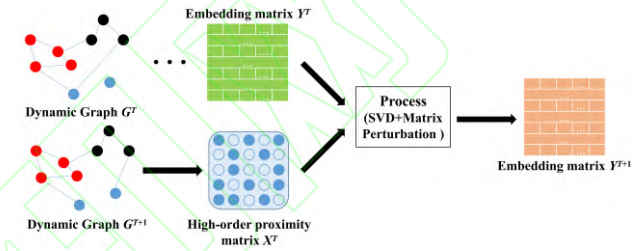


Fig.8 The framework of matrix factorization in dynamic graph embedding

图 8 基于矩阵分解的动态图嵌入模型框架

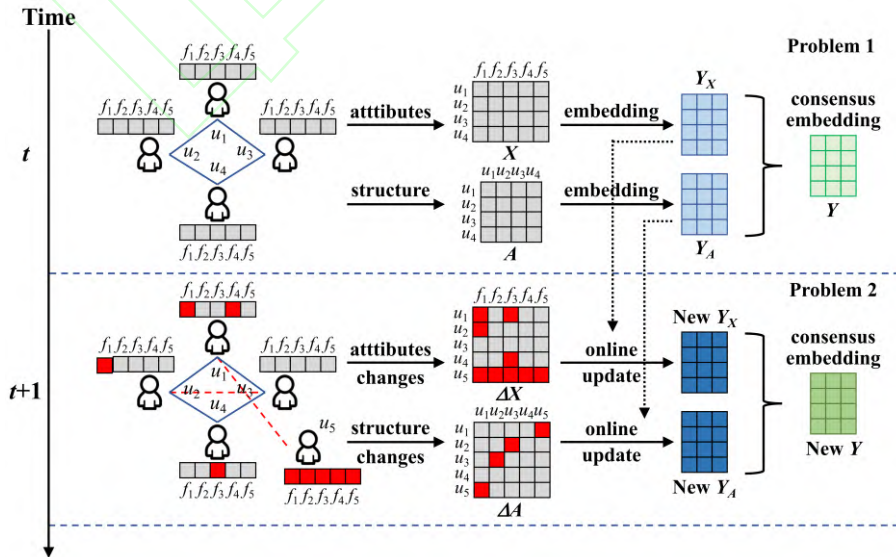


Fig.9 The framework of DANE^[36]. At time step t , DANE performs spectral embedding on graph structure \mathbf{A} and node attributes \mathbf{X} , and obtain embedding representation \mathbf{Y} . At time step $t+1$, DANE leverages matrix perturbation theory to update \mathbf{Y}_A and \mathbf{Y}_X , and give the new embedding \mathbf{Y}

图 9 DANE 模型结构^[36]。在 t 时刻, DANE 对拓扑结构和节点属性进行谱图嵌入, 并获得嵌入 \mathbf{Y} 。在 $t+1$ 时刻,

DANE 使用矩阵摄动理论更新 Y_A 和 Y_X , 获得新的嵌入 Y

DHPE^[38]同样采用分布式框架：静态部分，DHPE 与 HOPE 相似，将 GSVD 分解 Katz 相似度矩阵转换为广义特征值问题^[39]，使每个时刻生成的低维嵌入保留节点的高阶相似度；动态部分，DHPE 使用矩阵摄动理论更新 GSVD 的结果。此外，模型假设图中节点数恒定（添加或删除的节点为孤立节点），使每个时刻图的变化转化为边的变化。DHPE 能够在保持高阶相似性的同时更新节点嵌入，其增量计算方案有效提升了动态模型的计算效率。

Chen 等人^[40]提出了 TRIP 和 TRIP-BASIC 两种在线算法跟踪动态图的特征对，其核心思路是利用矩阵摄动理论对图的特征对进行更新。TRIP 和 TRIP-BASIC 引入图中三角形个数^[41,42]作为属性信息构建特征函数，然后将特征对映射成属性向量。上述模型能够有效追踪特征对、三角形个数和鲁棒性分数随时间的动态变化，但是模型的误差会随着时间推移不断积累。

由于增量矩阵的更新采用近似值的方式，导致生成嵌入的过程中误差不断积累。为解决上述问题，TIMERS^[43]采用 SVD 最大误差界重启算法，设置 SVD 重新启动时间，减少时间上的误差积累。该模型的核心包含两个部分：（1）增量更新，通过函数 f 近似地更新前一时刻的结果；（2）SVD 重启，通过设置误差阈值，确定执行 SVD 重启时刻，重新计算最优 SVD 结果，并最小化重启次数。

MF^[44]通过构造携带重要特征的矩阵因子，使潜在的 NMF 特征有效地表达动态信息。为了充分利用不同时刻的拓扑信息，MF 对嵌入空间的邻接关系矩阵进行整合，找到在各个时刻一致的嵌入矩阵 Y^* 和系数矩阵 W^* ，并通过最小化 Frobenius 范数使各时刻 Y^t 和 Y^* 、 W^t 和 W^* 差异最小。基于 NMF 加权表示相似性指数的 MF 比基于静态表示相似性指数的方法具有更好的性能。

DWSF^[45]采用使用 Semi-NMF^[46]和弱监督分解 (Weakly Supervised Factorization, WSF)^[47]，将数据矩阵 X 分解为标签矩阵 M 和嵌入矩阵 Y ($X = MY$,

其中 M 是非负的， X 和 Y 没有约束)，然后使用标签传播 (Label Propagation) 算法^[48]初始化标签矩阵因子，将类标签从有标签的数据实例传播到无标签的数据实例，最后将控制信息量的平滑度项与 Semi-NMF 相结合，优化模型参数生成嵌入。DWSF 将有限的监督信息合并为类别标签，在每次迭代中动态更新，大幅提升了模型在分类任务上的表现。

2.2 基于随机游走的图嵌入

受 word2vec^[15] 的启发，基于随机游走的图嵌入方法将节点转化为词，将随机游走序列作为句子，利用 Skip-Gram 生成节点的嵌入向量。随机游走法可以保留图的结构特性，并且在无法完整观察的大型图上仍有不错的表现。

2.2.1 基于随机游走的静态图嵌入

基于随机游走的静态图嵌入模型通过随机游走获得训练语料库，然后将语料库集成到 Skip-Gram 获得节点的低维嵌入表示。

Deepwalk^[49]使用随机游走对节点进行采样，生成节点序列，再通过 Skip-Gram 最大化节点序列中窗口 w 范围内节点之间的共现概率，将节点 v_j 映射为嵌入向量 Y_j (见图 10)：

$$\max \log \Pr(\{v_{j-w}, \dots, v_{j-1}, v_{j+1}, \dots, v_{j+w}\} | Y_j) \quad (7)$$

生成的嵌入 Y 将节点之间的关系编码在低维向量空间，用于捕捉邻域相似性和社区结构，学习节点的潜在特征。Deepwalk 不仅在数据量较少时有较好的表现，还可以扩展到大型图的表示学习。由于优化过程中未使用明确的目标函数，使得模型保持网络结构的能力受到限制。

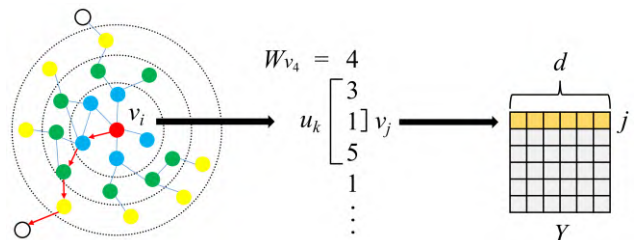


Fig.10 The framework of Deepwalk

图 10 Deepwalk 模型结构

node2vec^[50]在 Deepwalk 的基础上,引入有偏的随机游走(见图 11),增加邻域搜索的灵活性,生成质量更高、信息更多的嵌入表示。通过设置 p 和 q 两个参数,平衡广度优先搜索(Breadth-first Sampling, BFS)和深度优先搜索(Depth-first Sampling, DFS)策略,使生成的嵌入能够保持社区结构等价性或邻域结构等价性。虽然 node2vec 能够保持更多的一阶相似度和二阶相似度信息,但仍然缺少明确的目标函数来保持全局网络结构。

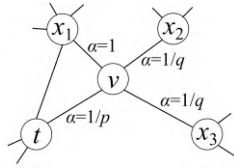


Fig.11 The random walk procedure in node2vec^[50]

图 11 node2vec 中的随机游走过程^[50]

Deepwalk 和 node2vec 采用随机游走探索节点局部邻域,使得学习到的低维表示无法保留图的全局结构,同时使用随机梯度下降求解非凸的目标函数,使生成的嵌入可能陷入局部最优解。为了解决上述问题, HARP^[51]将原始图中的节点和边递归地合并在一起,得到一系列结构相似的压缩图。这些压缩图有不同的粒度,提供了原始图全局结构的视图。从最粗略的形式开始,每个压缩图学习一组嵌入表示,并用于初始化下一个更细化的压缩图的嵌入。HARP 能够与 Deepwalk 和 node2vec 结合使用,提升原始模型性能,生成信息更丰富的嵌入表示。

利用分层 Softmax, Deepwalk 目标函数可以改写为矩阵分解的形式^[52]:

$$M_{ij} = \log \frac{[e_i(A + A^2 + \dots + A^k)]_j}{k} \quad (8)$$

式中, M_{ij} 是以节点 i 为起点 j 为终点的长度为 k 的路径期望值; e_i 是 one-hot 向量, 其第 i 个元素为 1 其余元素为 0, A^k 的不同幂次代表了不同的尺度。可见 Deepwalk 已经隐式地建模了从 1 到 k 阶的多尺度依赖关系, 但无法单独访问不同尺度。为了显式地构建 k 阶关系, Walklets^[53]修改了 Deepwalk 的采样过程, 在随机游走序列上跳过 $k-1$ 个顶点, 除

此之外,模型的优化和嵌入生成方式均与 Deepwalk 相同。相较于 Deepwalk, Walklets 能够捕获节点与社区之间不同尺度的层次结构,进而显式建模对多尺度关系,保留更丰富的节点从属关系信息。

TriDNR^[54]是首个利用标签信息进行表示学习的深层神经网络模型,能够同时利用网络结构、节点特征和节点标签学习节点嵌入表示(见图 12)。对于网络结构,通过随机游走序列最大化共现概率来保持图中节点间的邻近关系;对于节点特征,通过最大化给定节点的单词序列的共现概率捕获节点与单词的相关性;对于节点标签,通过最大化给定类别标签的单词序列的概率建模标签与单词的对应关系。最后,使用耦合神经网络的算法将三部分信息合并为节点嵌入。

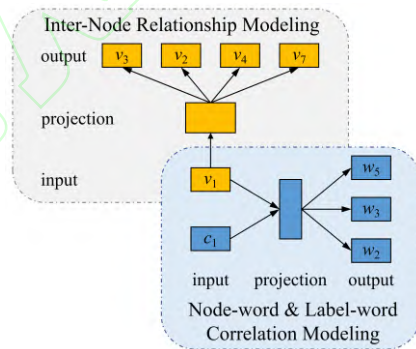


Fig.12 Architecture of the TriDNR model^[54]. It couples two neural networks to capture the inter-node, node-word, and label-word relationship

图 12 TriDNR 模型结构^[54], 使用两个神经网络来捕获节点—节点、节点—单词、标签—单词之间的关系

2.2.2 基于随机游走的动态图嵌入

基于随机游走的动态图嵌入模型将每条边与对应时刻相关联,使随机游走序列由一系列包含递增时刻的边所连接的节点构成,最后利用 Skip-Gram 模型将每个节点映射成 d 维向量。

Sajjad 等人^[55]将图嵌入的生成过程分为两步:首先,更新动态图上的随机游走序列。与直接在静态快照上从头开始随机游走相比,更新算法保持了随机游走的统计特性。然后,在给定前一时刻的嵌入表示以及更新后的随机游走序列的条件下,利用

Skip-Gram 模型对嵌入表示进行更新。CTDNE^[56]则利用时间随机游走从连续型动态图中学习包含时间信息的嵌入表示。实际上，CTDNE 采用的时间随机游走与静态图方法相似，但约束每个随机游走符合边出现的时间顺序，即边的遍历必须按照时间递增的顺序，由于每条边包含多个时间戳，使得同一节点可能在游走中出现多次。时间信息的引入减少了嵌入的不确定性，使 CTDNE 在众多任务上的表现优于 Deepwalk 和 node2vec 等静态模型。

在动态图中应用静态方法存在两大问题：（1）对每个快照都进行随机游走非常耗时；（2）不同快照的嵌入空间并不一致。为解决上述问题，Mahdavi 等人在 node2vec 的基础上，提出了动态图嵌入模型 dynnode2vec^[57]。该模型在快照 G^1 上运行 node2vec 获得嵌入向量以及训练后的 Skip-Gram 模型。对于后续快照，在两个连续快照之间执行以下步骤：（1）为演化节点生成随机游走序列。（2）使用动态 Skip-Gram^[58]将前一时刻学习到的嵌入作为初始权值，结合演化节点的随机游走生成当前时刻嵌入。由于动态图是逐渐演化的，即大多数节点的邻域保持不变，dynnode2vec 仅对演化节点进行随机游走大幅提升了模型效率。

STWalk^[59]是一种无监督节点轨迹学习算法，通过捕捉给定时间窗口内节点变化生成嵌入表示。该模型将当前时刻快照上的随机游走定义为空间游走，过去时刻快照上的随机游走定义为时间游走，从而捕捉节点的时空行为，然后利用 Skip-Gram 生成节点轨迹的嵌入表示。STWalk 有两种不同的变体：STWalk1 同时考虑空间游走和时间游走来学习嵌入表示；STWalk2 将空间游走和时间游走建模为两个子问题并分别求解，然后组合两个结果获得最终的嵌入表示。上述模型仅使用图的结构信息学习捕获节点轨迹时空特性的低维嵌入，未考虑节点特征和标签信息。

Deepwalk 和 node2vec 模仿单词嵌入作为节点嵌入表示，而 tNodeEmbed^[60]模仿句子嵌入^[61]作为节点嵌入表示。句子中每个单词不仅代表节点随时间变化的向量，还捕捉了节点角色和连接关系的动

态变化。为此，tNodeEmbed 使用联合损失函数优化两个目标：（1）三维特征空间中节点的静态邻域；（2）图的动态特性。tNodeEmbed 使用 node2vec 对节点嵌入进行初始化，将不同时刻的节点表示进行对齐，然后根据给定的图分析任务和过去时刻的节点嵌入联合学习，使生成的嵌入既保留图的结构和动态信息，又适用于特定任务。

2.3 基于自编码器的图嵌入

自编码器(Autoencoder, AE)^[62]是一种人工神经网络，包含编码器和解码器两个部分，用于无监督地构造输入数据的向量表示。通过对数据中的非线性结构进行建模，自编码器使隐藏层学习到的表示维度小于输入数据，即对原始数据进行降维。基于自编码器的图嵌入方法使用自编码器对图的非线性结构建模，生成图的低维嵌入表示。

2.3.1 基于自编码器的静态图嵌入

基于自编码器的图嵌入方法起源于使用稀疏自编码器的 GraphEncoder^[63]。其基本思想是将归一化的图相似度矩阵作为节点原始特征输入到稀疏自编码器中进行分层预训练，使生成的低维非线性嵌入可以近似地重建输入矩阵并保留稀疏特性：

$$\begin{aligned} \text{Encoder: } \mathbf{Y} &= f(\mathbf{X}) \\ \text{Decoder: } \hat{\mathbf{X}} &= g(\mathbf{Y}) \end{aligned} \quad (9)$$

式中， $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ 表示输入矩阵， $\hat{\mathbf{X}} = \{\hat{\mathbf{x}}_i\}_{i=1}^N$ 表示重构的输出矩阵， $f(\cdot)$ 是编码器， $g(\cdot)$ 是解码器。简言之，GraphEncoder 将输入矩阵 \mathbf{X} 的信息压缩到低维嵌入 \mathbf{Y} 中，并利用 L2 重建损失对 \mathbf{Y} 进行优化。稀疏自编码器降低了计算复杂度，相较之前常用的谱聚类方法，具有灵活高效的特性。

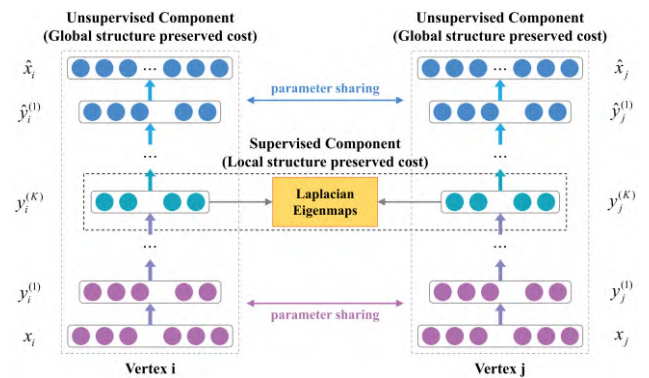


Fig.13 The framework of SDNE^[64]. Both the first and second-order proximities of nodes are preserved by supervised and unsupervised components respectively

图 13 SDNE 模型结构^[64]。有监督组件和无监督组件分别保持节点的一阶和二阶相似度

SDNE^[64]利用深度自编码器以及图的一阶、二阶相似度,捕获高度非线性的网络结构。SDNE 包含有监督组件和无监督组件(见图 13)。有监督组件引入拉普拉斯特征映射作为一阶相似度的目标函数,使生成的嵌入捕获局部结构信息。无监督组件修改 L2 重建损失函数作为二阶相似度的目标函数,使生成的嵌入捕获全局结构信息。对一阶和二阶相似度联合优化,增强了模型在稀疏图上的鲁棒性,使生成的嵌入同时保留全局和局部结构信息。

DNGR^[65]生成低维嵌入的过程主要分为三步:

(1) 利用随机冲浪模型捕捉图的结构信息,并生成共现概率矩阵;(2) 利用共现概率矩阵计算正逐点互信息(Positive Pointwise Mutual Information, PPMI)矩阵^[66];(3) 将 PPMI 矩阵输入堆叠去噪自编码器(Stacked Denoising Autoencoder, SDAE)生成低维嵌入表示。相较于随机游走,随机冲浪直接获取图的结构信息,克服了原有采样过程的限制;PPMI 矩阵能够保留图的高阶相似度信息;堆叠结构使隐层维度平滑递减,提升深层架构学习复杂特征的能力,同时去噪策略增强了模型的鲁棒性。

DNE-APP^[67]利用半监督堆叠式自编码器(Stacked Autoencoder, SAE)生成保留 k 阶信息的低维嵌入主要分为两步:(1) 使用 PPMI 度量和 k 步转移概率矩阵,生成包含 k 阶信息的相似度聚合矩阵。(2) 使用 SAE 重构相似度聚合矩阵,学习低维非线性嵌入表示。与仅保持一阶和二阶相似度的 SDNE 相比,DNE-APP 模型可以保持不同的 k 阶相似度;与仅重建高阶相似度的 DNGR 相比,DNE-APP 在重建过程中引入了成对约束,使相似节点在嵌入空间更加接近。

变分自编码器(Variational Autoencoder, VAE)^[68]是用于降维的生成式模型,其优势为容忍噪声和学习平滑的表示。VGAE^[69]首先引入 VAE 学习可解释

的无向图嵌入表示。模型的编码器部分使用图卷积网络(Graph Convolutional Network, GCN)^[70]:

$$q(Y|X, A) = \prod_{i=1}^N q(y_i|X, A), \quad (10)$$

$$\text{with } q(y_i|X, A) = N(y_i|\mu_i, \text{diag}(\delta_i^2))$$

式中, $\mu = \text{GCN}_\mu(X, A)$ 是均值向量 μ_i 的矩阵, 同样 $\log \delta = \text{GCN}_\delta(X, A)$ 为方差向量。模型的解码器部分使用嵌入的内积:

$$p(A|Y) = \prod_{i=1}^N \prod_{j=1}^N p(A_{ij}|y_i, y_j), \quad (11)$$

$$\text{with } p(A_{ij}=1|y_i, y_j) = \sigma(y_i^T y_j)$$

式中, $\sigma(\cdot)$ 是 sigmoid 函数。最后, VGAE 通过最小化重构损失和变分下界对模型进行优化:

$$L = E_{q(Y|X, A)}[\log p(A|Y)] - \text{KL}[q(Y|X, A) \| p(Y)] \quad (12)$$

式中, $\text{KL}[q(\cdot) \| p(\cdot)]$ 为 $q(\cdot)$ 和 $p(\cdot)$ 的 KL 散度^[71], $p(Y)$ 为高斯先验。

Salha 等人提出的 Linear-VGAE 模型^[72]使用基于归一化邻接矩阵的简单线性模型替换 VGAE 中的 GCN 编码器:

$$\mu = \tilde{A}XW_\mu \text{ and } \log \sigma = \tilde{A}XW_\sigma \quad (13)$$

式中, \tilde{A} 为归一化的邻接矩阵, W_μ 和 W_σ 为权重矩阵。Linear-VGAE 仅使用图的直接邻域及两个权重矩阵生成嵌入维表示, 未使用激活函数, 降低了编码器复杂度。通过对比 VGAE 模型性能, 证明简单的线性节点编码方案的同样有效。

与一般的非对称模型不同, GALA^[73]采用完全对称的图卷积自编码器模型生成图的低维嵌入表示。在对输入矩阵重构的过程中, 编码器执行的拉普拉斯平滑^[74]与解码器的拉普拉斯锐化^[75]相对称。与现有的 VGAE 方法不同, 为了使解码器可以直接重构节点的特征矩阵, GALA 使用谱半径为 1 的拉普拉斯锐化表示。相较于仅使用 GCN 编码器的模型, GALA 的对称结构, 能够在编码和解码过程中同时使用结构信息和节点特征。

ANE^[76]使用对抗性自编码器^[77]生成捕获高度非线性结构信息的低维嵌入。具体而言, ANE 利用一阶和二阶相似度捕捉图的局部和全局结构, 使生成的嵌入可以保持高度的非线性, 同时训练过程对抗性自编码器的两个准则: 一是基于重建误差的自

编码器训练准则，二是将嵌入表示的聚合后验分布与任意先验分布匹配的对抗性训练准则。通过施加对抗性正则化，ANE 改善了嵌入生成过程中的流形断裂问题，提升了低维嵌入的表示能力。

2.3.2 基于自编码器的动态图嵌入

基于自编码器的动态图方法将每个时刻训练的参数作为下一时刻自编码器的初始值，从而在一定程度上保持生成嵌入的稳定性，节省模型的训练时间，提高模型的计算效率。

受 SDNE 的启发，Goyal 等人提出了快照型动态图嵌入模型 DynGEM^[78]（见图 14）。该模型使用深度自编码器将输入数据映射到高度非线性的低维嵌入空间，以捕捉任一时刻快照中节点的连接趋势，同时下一个时刻的嵌入模型直接继承前一个时刻的训练参数，使 t 时刻的快照嵌入可以利用 $t-1$ 时刻快照嵌入进行增量地学习。由于动态图中节点的数量不断变化，DynGEM 设计了一种可动态调节神经网络中神经元个数、隐层层数的方法 PropSize。在训练过程中，DynGEM 结合一阶相似度和二阶相似度保留局部结构信息和全局结构信息，同时引入 L1 和 L2 正则化进一步提升模型性能。需要注意的是 DynGEM 不强加保持相邻时刻嵌入接近的显式正则化，即相邻时刻的快照如果明显不同，则相应

的编码函数 f_{t-1} 和 f_t 也有所不同。

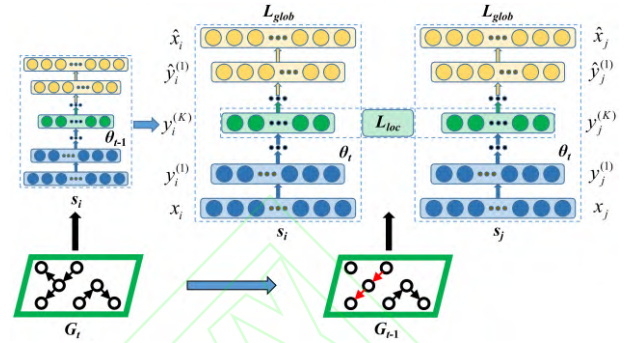


Fig.14 The framework of DynGEM^[78]

图 14 DynGEM 模型结构^[78]

DynGEM 生成当前时刻嵌入时只捕获了前一时刻的信息，致使大量历史信息被忽略，为此 Goyal 等人提出了另一个基于自编码器的动态图嵌入模型 dyngraph2vec^[79]。该模型将之前 l 个时刻的图结构信息作为输入，将当前时刻生成图嵌入作为输出，从而捕获当前时刻与之前多个时刻节点之间的非线性交互信息。该模型有三种变体（见图 15）：dyngraph2vecAE 以一种简单的方式对自编码器进行扩展；dyngraph2vecRNN 和 dyngraph2vecAERN 使用长短期记忆网络 (Long Short-Term Memory, LSTM)^[80] 对历史信息编码。在动态图的演化过程中，dyngraph2vec 仅使用相邻接点，未考虑图的高阶相似度信息。

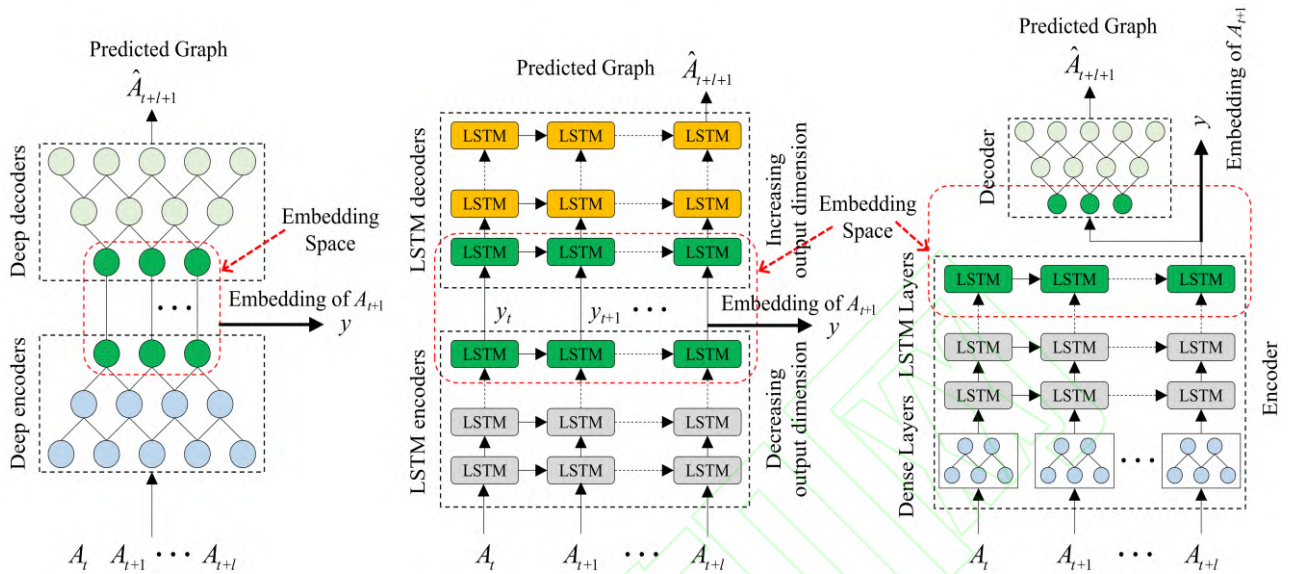


Fig.15 Variations of dyngraph2vec^[79]: dyngraph2vecAE, dyngraph2vecRNN and dyngraph2vecAERNN

图 15 dyngraph2vec 变体结构^[79]: dyngraph2vecAE、dyngraph2vecRNN 和 dyngraph2vecAERNN

随着动态图的演化, NetWalk^[81]可以增量地学习网络表示(见图 16)。具体而言, NetWalk 利用初始的动态图上提取的多个游走序列以及深度自编码器的隐藏层生成节点嵌入表示。在训练过程中, NetWalk 联合最小化游走序列中成对节点的表示距离和自编码器的重构误差,使学习到的嵌入表示既可以实现局部拟合,又可以实现全局正则化。NetWalk 对生成到的嵌入表示使用动态聚类模型,能够标记图中异常^[82,83]的节点或边。

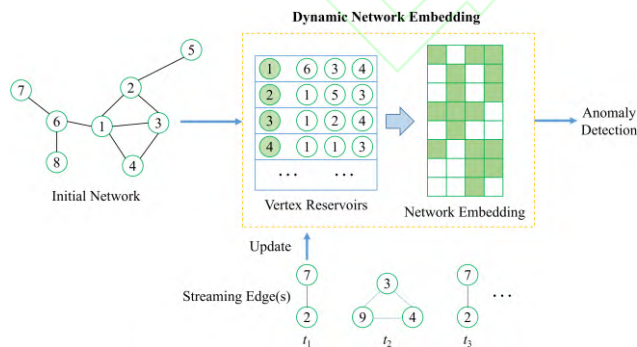


Fig.16 Workflow of NetWalk for anomaly detection^[81]

图 16 NetWalk 异常检测的流程^[81]

BurstGraph^[84]将动态图的演化分为一般演化和突发演化,并使用两个基于 RNN 的 VAE 分别对每个时刻的演化信息进行建模。在编码器部分,两个自编码器共同使用 GraphSAGE^[85]学习到的节点

特征和拓扑结构信息。对于突发演化, BurstGraph 在 VAE 中引入了 spike & slab 分布^[86]作为近似后验分布;对于一般演化, BurstGraph 使用原始的 VAE 模型。为了充分利用图的动态信息, BurstGraph 使用 RNN 捕捉每个时刻的图结构,将一般演化和突发演化信息保留在 RNN 状态单元中,并随时间的推移不断更新状态单元。由于生成的嵌入中保留了突发信息, BurstGraph 常用于关于图的异常检测中的突发检测^[87,88]任务。

动态图嵌入通常存在三个方面的局限性: (1) 嵌入表示空间^[89,90], 欧式空间表示可能导致图的潜在层次结构失真; (2) 动态信息, 忽视图的动态演化通常会导致模型错误地利用未来信息来预测过去的交互, (3) 不确定性^[91], 图的固有特性, 生成的确定性表示不能对不确定性建模。为了解决上述问题, HVGNN^[92]采用双曲变分 GNN 对双曲空间中的动态图进行建模, 使生成的嵌入同时包含图的动态信息和不确定性。具体而言, HVGNN 使用双曲空间代替欧式空间, 同时引入新的时间 GNN(Temporal GNN, TGNN)来建模动态特性。为了建模图的不确定性, HVGNN 设计了一个基于 TGNN 的双曲 VGAE, 使模型可以对不确定性和动态进行联合建模。最后, 引入的重参数化采样算法,

实现模型的梯度学习。相较于欧式空间计算量的指数增长，双曲空间降低了模型的计算复杂度；对于不确定性的建模，增强了传递较少信息且具有较多动态特性节点的表示性能。

2.4 基于图神经网络的图嵌入

图神经网络(Graph Neural Network, GNN)是专门处理图数据的深度模型，其利用节点间的消息传递来捕捉某种依赖关系，使生成的嵌入可以保留任意深度的邻域信息。由于 GNN 模型强大的表示能力，嵌入模型的性能得到了显著提升。

2.4.1 基于图神经网络的静态图嵌入

基于 GNN 的静态图模型聚合节点邻域的嵌入并不断迭代更新，利用当前的嵌入及上一次迭代的嵌入生成新的嵌入表示。通过多次迭代，GNN 模型学习到的嵌入可用于表征全局结构。

GCN 使用节点特征矩阵 X 与邻接矩阵 A 生成包含节点的特征信息的嵌入表示（见图 17）。对于多层 GCN，层间传播公式为：

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}) \quad (14)$$

其中， $\tilde{A} = A + I$ ， \tilde{D} 为 \tilde{A} 的度矩阵， $\sigma(\cdot)$ 为激活函数， $H^{(l)}$ 为各层激活矩阵，且 $H^{(0)}$ 为 X 。为了提高层间传播效率，GCN 使用图上谱卷积一阶近似^[93,94]。对于双层 GCN 模型，其前向传播公式为：

$$Y = f(X, A) = \sigma^{(1)}(\hat{A} \sigma^{(0)}(\hat{A} X W^{(0)}) W^{(1)}) \quad (15)$$

其中， $\hat{A} = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}}$ 。最终，GCN 使用神经网络模型 $f(X, A)$ 对图结构进行编码，生成高质量的嵌入表示。GCN 的主要缺点是扩展性相对较差，因为训练过程执行全批次梯度下降收敛较慢，导致模型难以扩展到大规模网络。

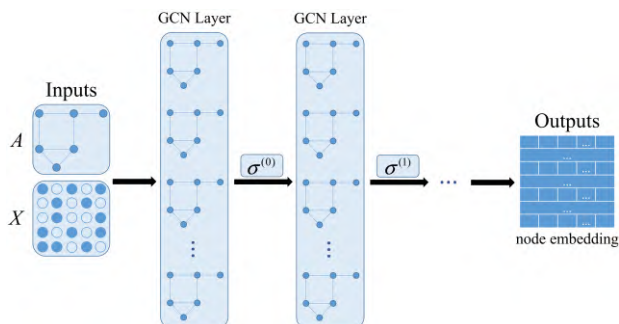


Fig.17 The framework of multi-layer GCN. The graph structure is shared over layers

图 17 多层 GCN 结构。结构信息在各层之间共享

GCN 通常应用于固定图的直推式表示学习，GraphSAGE 将其扩展到归纳式无监督学习的任务中，利用节点特征（例如，文本属性、节点度）学习不可见节点的嵌入表示。GraphSAGE 不是为每个节点训练单独的嵌入，而是通过采样和聚合节点的局部邻域特征训练聚合器函数，同时学习每个节点邻域的拓扑结构及特征分布，生成嵌入表示（见图 18）。相比 GCN，GraphSAGE 使用无监督损失函数强制邻近节点具有相似表示，远距离节点具有不同的表示。在给定下游任务的情况下，GraphSAGE 能够替换或增加相应的目标函数，进行有监督或半监督学习，提升模型的灵活性；训练过程执行分批次训练，提升模型的收敛速度。

图注意力网络(Graph Attention Network, GAT)^[95]在 GCN 的基础上引入注意力机制^[96]，对邻近节点特征加权求和，分配不同的权值。针对单个节点，GAT 使用 self-attention^[97]获得注意力系数：

$$e_{ij} = a(W \vec{h}_i, W \vec{h}_j) \quad (16)$$

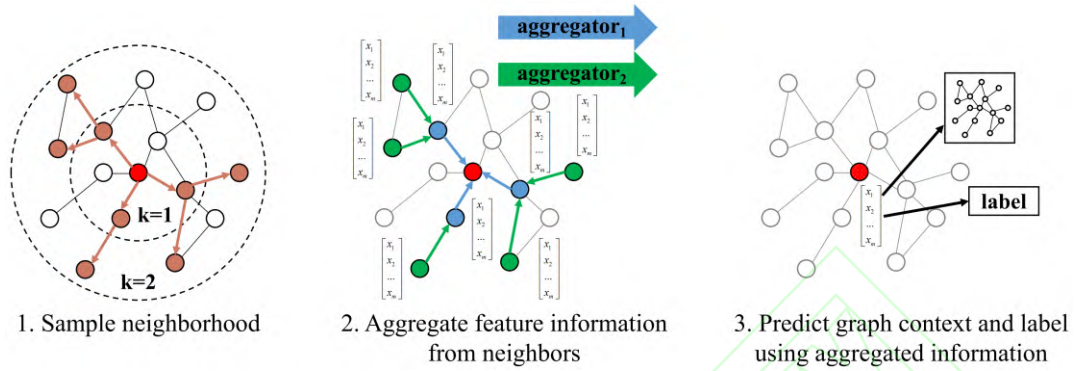
e_{ij} 表示的是节点 j 对节点 i 的重要性。GAT 使用 masked-attention^[97]将注意力分配到节点 i 的邻域：

$$\alpha_{ij} = \frac{\exp(\text{LeakyReLU}(\vec{a}^T [W \vec{h}_i \| W \vec{h}_j]))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(\vec{a}^T [W \vec{h}_i \| W \vec{h}_k]))} \quad (17)$$

式中， \vec{a} 是层间的权重矩阵， $\|$ 表示拼接运算。最后，使用 multi-head attention^[97]生成节点嵌入：

$$y_i = \sigma\left(\frac{1}{K} \sum_{k=1}^K \sum_{j \in N_i} \alpha_{ij}^k W^k \vec{h}_j\right) \quad (18)$$

式中， σ 表示激活函数， K 表示 multi-head attention 的头数， α^k 为第 k 个 self-attention。注意力参数全图共享，大幅减少了参数占用的存储空间，同时 GAT 对所有邻居节点进行采样，较 GraphSAGE 得到的嵌入更具表征性。GAT 的缺点在于使用二阶以上邻居时，容易导致过度平滑。

Fig.18 Visual illustration of the GraphSAGE sample and aggregate approach^[85]图 18 图解 GraphSAGE 采样和聚合方式^[85]

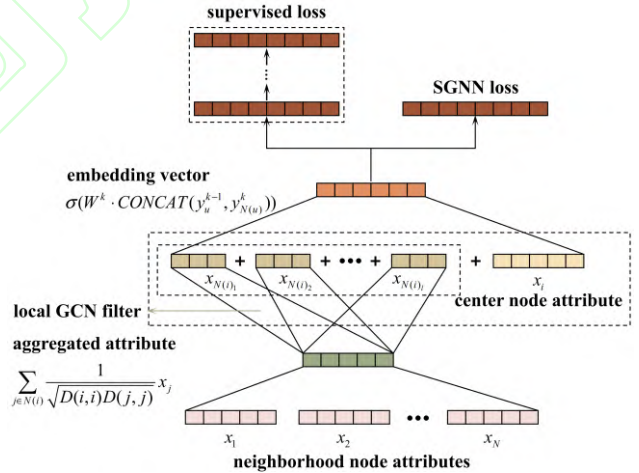
用于图嵌入的 GNN 模型大多遵循邻域聚合架构,通过递归地聚合和变换邻域节点的特征向量生成节点嵌入,因此不能有效区分某些简单的图结构。为了提高 GNN 模型对图结构的辨识能力,图同构网络(Graph Isomorphism Network, GIN)^[98]利用 GNN 和 Weisfeiler-Lehman(WL)图同构测试^[99]之间的密切联系,使生成的嵌入保留图结构辨识信息。WL 测试通过聚合给定节点邻居的特征向量对该节点进行迭代更新,同时使用内射聚合更新将不同的节点邻域映射为不同的特征向量。GIN 采用与 WL 内射聚合相似的方式进行建模:首先,将节点邻居的特征向量抽象为一个多集;然后,将邻居聚合运算抽象为多集上的函数(多集函数的区分性越强底层的表征能力就越强);最后,将生成的嵌入用于图分类任务,其性能可以匹配 WL 测试的结果。

MF-GCN^[100]是一种多滤波 GCN 模型(见图 19),在每个传播层使用多个局部 GCN 滤波器进行特征提取,使模型捕捉到节点特征的不同方面。对于第一层,MF-GCN 对节点属性进行如下操作:

$$\mathbf{x}_{N(i)_l} = \alpha \left(\sum_{j \in N(i)} \frac{1}{\sqrt{D(i,i)D(j,j)}} \mathbf{x}_j \mathbf{W}_l \right) \quad (19)$$

式中, \mathbf{x}_j 是节点 j 的特征向量, $\mathbf{x}_{N(i)_l}$ 表示第 l 个局部 GCN 生成的节点 i 邻域低维嵌入, α 为激活函数。将多个局部 GCN 生成的表示进行拼接生成节点邻

域嵌入表示 $\mathbf{y}_{N(u)}^k$, 再与 $k-1$ 层的嵌入 \mathbf{y}_u^{k-1} 拼接即为第 k 层嵌入 \mathbf{y}_u^k 。MF-GCN 使用的局部 GCN 和优化策略与 GraphSAGE 相似,关键的不同在于 MF-GCN 使用多个局部 GCN 获取节点特征不同方面的信息,使生成的嵌入保留更丰富的属性信息。

Fig.19 The framework of MF-GCN^[100]图 19 MF-GCN 模型结构^[100]

多数 GCN 模型中邻域相互作用项的系数相对较小,导致性能与采用线性策略的简化图卷积网络(simplified graph convolutional networks, SGC)^[101]相当。为了有效捕捉图的复杂非线性, GraphAIR^[102]同时对邻域聚合和邻域交互进行建模。GraphAIR 由两部分组成:邻域聚合模块通过组合邻域特征来构建节点表示;邻域交互模块通过乘法运算显式建模邻域交互。现有的大多数 GCN 模型与 GraphAIR 兼容,可以提供即插即用的邻域聚合模块和邻域交

互模块。此外, GraphAIR 利用残差学习策略, 将邻域交互与邻域聚合分离, 使模型更容易优化。

SDGNN^[103]是用于处理符号有向图的嵌入模型, 其在传统 GNN 的基础上考虑了平衡理论^[104]和地位理论^[105], 重新设计聚合器和损失函数。SDGNN 聚合来自不同邻域的信息, 并使用 MLP(Multilayer Perceptron)将这些消息编码为节点嵌入。SDGNN 的聚合器可分为两类: 一是平均聚合器, 二是注意力聚合器。为了优化生成的嵌入, SDGNN 使用组合损失函数来重构网络中的符号、方向和三角形三个关键特征。平衡理论和地位理论的引入, 使 SDGNN 在考虑边缘符号的同时兼顾方向信息, 提升了模型在符号图分析任务中的表现。

2.4.2 基于图神经网络的动态图嵌入

基于 GNN 的动态图模型通常在静态图模型的基础上, 引入一种循环机制更新网络参数, 实现动态过程的建模, 使生成低维嵌入可以有效保留图的动态演化信息。

DyRep^[106]将动态图嵌入假设为图的动态(拓扑演化)和图上的动态交织演化(节点间的活动)的中介过程。DyRep 采用关联和通信事件的形式接收动态信息, 并基于以下原则捕获观察到的事件的影响, 更新有关节点的表示: (1) 自我传播。自我传播是支配单个节点演化的动力中最小的组成部分。节点相对于其先前位置在嵌入空间中演化, 而

不是以随机方式进化。(2) 外源驱动。一些外力可以在某时间间隔平滑地更新该节点的当前特征。

(3) 局部嵌入传播。事件中涉及的两个节点形成临时(通信)或永久(关联)路径, 用于信息从一个节点的邻域传播到另一个节点。DyRep 采用双时间尺度来捕捉图级和节点级的动态过程, 计算节点表示并参数化。最后, 使用时间注意机制^[95,107]耦合模型的结构和时间信息, 使生成的节点嵌入可以捕获非线性的动态信息。DyRep 作为一种归纳式学习模型, 强调学习节点的表示方法而不是节点的固定表示, 因此更利于新的节点表示的生成。

DySAT^[108]通过邻域结构和时间两个维度的联合自注意力来计算节点嵌入, 主体为三个模块(见图 20): (1) 结构注意力块通过自注意聚集和堆叠从每个节点局部邻域中提取的特征, 以计算每个快照的中间表示并输入到时间模块; (2) 时间自注意力块通过嵌入每个快照的绝对时间位置^[109]来捕获排序信息, 并将位置嵌入与结构注意力块的输出组合, 输入到前馈层; (3) 图上下文预测模块通过跨多个时间步长的目标函数保留节点的邻域信息, 使生成的嵌入能够捕捉结构演化。DySAT 的优点在于使用多头注意力能够捕获动态性的多个方面, 缺点在于该模型仅适用于节点数恒定的动态图, 并且节点共现率作为损失函数导致模型捕捉节点的动态变化的能力有限。

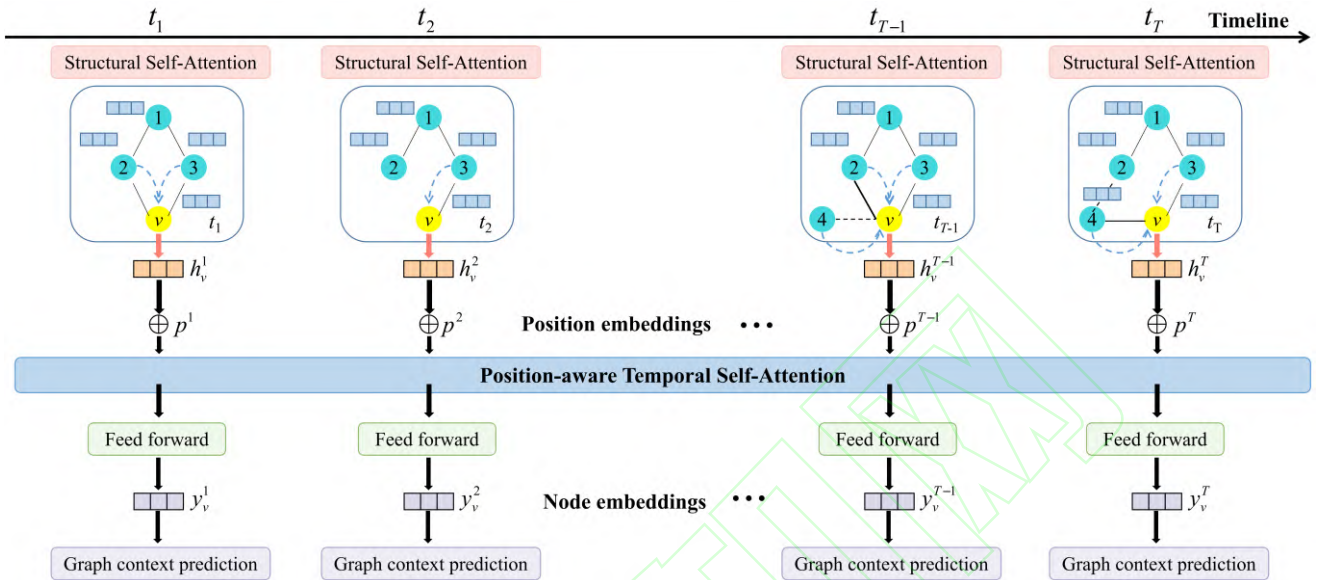


Fig.20 The framework of DySAT^[108]: structural attention block, temporal attention block, and context prediction

图 20 DySAT 模型结构^[108]: 结构注意力模块、时间注意力模块和上下文预测

动态图中节点可能频繁出现和消失,使得 RNN 在学习这种不规则的行为时非常具有挑战性。为了解决这个问题, EvolveGCN^[110]在每个时间步使用 RNN 来调整 GCN(即网络参数),即关注 GCN 参数在每个时刻的演化而不关注该时刻的节点表示,这不仅提高了模型的自适应性和灵活性,还可以保持图的动态信息。此外, EvolveGCN 只对 RNN 参数进行训练,不再训练 GCN 参数,这种方式使得参数的数量不会随着时刻的增加而增长,使该模型像常用的 RNN 一样易于管理。

在新的边出现时, DGNN^[111]不仅更新节点表示,同时将交互信息传播到其他受影响的节点,使嵌入信息在更新和传播过程中可以合并交互之间的时间间隔。当新的边出现时,两端节点及其一阶邻域都有明显的概率变化。此外,邻域受到的影响与时刻有关,最近时刻与端点交互的邻居节点对出现的新变化很敏感,而较远的过去时刻的邻居受影响较小。端点和一阶邻域均使用时态信息增强 LSTM 作为更新模块的基本框架,使模型能够处理不同时间间隔的信息。

TemporalGAT^[112]通过集成 GAT 和时间卷积网络(Temporal Convolutional Network, TCN)^[113,114]来

学习动态图上的嵌入表示(见图 21)。该模型将 self-attention 应用于节点邻域,并通过 TCN 保留图的动态信息。最后,采用二元交叉熵损失函数^[115]学习节点表示,并预测节点之间是否存在边。TemporalGAT 不仅能够建模节点数目不固定的动态图,还能同时捕获动态图的结构信息与时间信息。

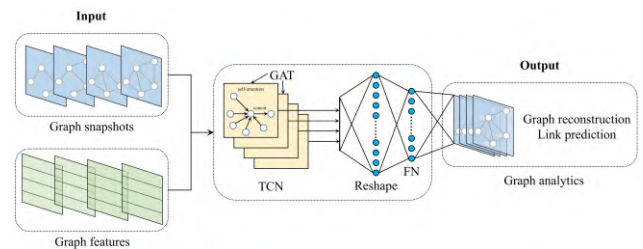


Fig.21 The framework of TemporalGAT^[112]

图 21 TemporalGAT 模型结构^[112]

2.5 基于其他方法的图嵌入

2.5.1 基于其他方法的静态图嵌入

LINE^[116]同样定义了一阶相似度和二阶相似度函数,并对其进行优化。一阶相似度用于保持邻接矩阵和嵌入表示的点积接近,二阶相似度用于保持上下文节点的相似性。为了结合一阶和二阶相似

度, LINE 分别优化一阶和二阶相似度的目标函数, 然后将生成的嵌入向量进行拼接。LINE 的边采样策略克服了随机梯度下降的局限性, 使其能够应用到大规模图嵌入中; 但是一阶和二阶表示单独优化以及简单的拼接操作, 限制了模型表示能力。

DRNE^[117]没有重建邻接矩阵, 而是直接使用 LSTM 聚合邻域信息重建节点嵌入 (见图 22)。DRNE 的目标函数如下: 由于 LSTM 要求其输入是一个序列, 所以 DRNE 根据节点的度数进行排序, 同时对度数较大的节点采用邻域抽样技术, 以防止过长的记忆。这种方法可以保持节点的正则等价性和多种中心性度量 (如 PageRank^[118])。

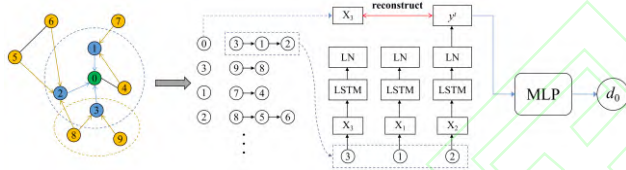


Fig.22 The framework of DRNE^[117]

图 22 DRNE 模型结构^[117]

Transformer^[97]架构已经成为许多领域的主流选择, 但在图级预测任务中, 通常表现不佳。为此, Ying 等人在标准 Transformer 的基础上利用图的结构信息构建 Graphormer^[119] (见图 23)。对于结构信息的编码主要分为三部分: (1) 中心性编码 (Centrality Encoding) 用于捕捉图中节点的重要性, 根据每个节点的度为每个节点分配一个可学习向量, 并将其添加到输入层的节点特征中。(2) 空间编码 (Spatial Encoding) 用于捕捉节点之间的结构关系, 根据每个节点对的空间关系为其分配了一个可学习的嵌入。(3) 边编码 (Edge Encoding) 用于捕捉边缘特征中额外的空间信息, 然后将其输入到 Transformer 层。Graphormer 同时使用上述结构信息编码, 提升模型性能, 进而生成更优的嵌入表示。

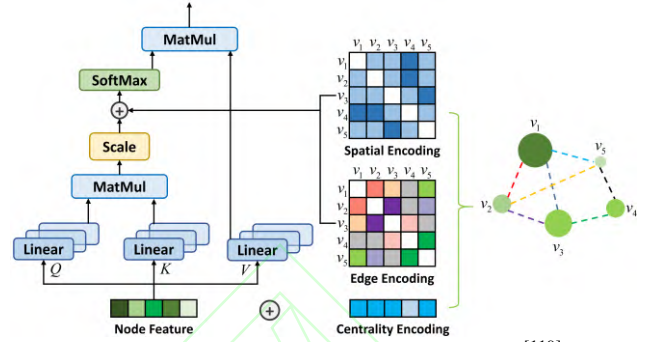


Fig.23 The framework of Graphormer^[119]

图 23 Graphormer 模型结构^[119]

2.5.2 基于其他方法的动态图嵌入

HTNE^[120]使用 Hawkes 过程^[121]捕捉历史邻域对当前邻域的影响, 建模动态图中节点的邻域序列。然后, 将节点分别映射为基向量和历史向量, 并输入到 Hawkes 过程以生成嵌入表示。由于历史邻域对当前邻域形成的影响因节点而不同, 引入注意力机制调节影响的大小。HTNE 的核心在于使用邻域的形成过程描述节点的动态演化, Hawkes 过程及注意力的引入使生成的嵌入有效集成到上述信息。

DynamicTriad^[122]通过施加三元组 (一组三个顶点的集合) 模拟图的动态变化。一般来说, 三元组分为两种类型: 闭合三元组和开放三元组。由开放三元组演化为封闭三元组的三元组闭合过程是动态图形成和演化的基本机制^[123,124]。DynamicTriad 采用统一的框架来量化上述闭合过程, 使模型能够有效捕捉图的动态信息。

动态图通常是在微观和宏观两个方面随时间演化, 微观动态可以描述拓扑结构的形成过程, 宏观动态描述了图规模的演化模式。M²DNE^[125]是首个将微观动态和宏观动态同时融入到动态图嵌入过程的模型。对于微观动态, M²DNE 把边的建立当作时间事件, 并提出时间注意点流程来捕获用于嵌入生成的时间属性。对于宏观动态, M²DNE 通过定义使用嵌入参数化的动态方程来捕获内在演化模式, 再利用内在演化模式在更高层次上约束图的拓扑结构。最后, M²DNE 利用微观动态和宏观动态的演化和交互, 生成节点嵌入。微观动态描述

的是网络结构的形成过程,宏观动态描述的是网络规模的演化模式。大多数动态图方法只考虑了微观动态,忽略了宏观动态在保持网络结构和演化模式方面的重要价值, M^2DNE 同时使用宏观动态和微观动态,进而增强了模型的泛化能力。

因果匿名游走 (Causal Anonymous Walks, CAW)^[126]与匿名游走 (Anonymous Walks, AW)^[127]相比有两个不同性质: (1) 因果关系提取。CAW 从感兴趣的链接开始,随着时间的推移回溯多个相邻链接,以编码动态图的基本因果关系。(2) 基于集合的匿名化。CAW 删除遍历集合中的节点标识以保证归纳学习,同时根据特定位置出现的计数对相应节点标识进行编码。CAW-N 是专门用于链接预测的变体,该模型对两个感兴趣的节点进行 CAW 采样,然后通过 RNN 和集合池化对采样结果进行编码和聚合,生成最终嵌入。CAW-N 不仅保留了游走过程中所有细粒度的时间信息,还可以通过 motif^[128,129]计数将其移除。

2.6 图嵌入模型关系分析

图嵌入可以解释为生成图数据的向量表示,用于洞察图的某种特性。表 2 和表 3 归纳了主要静态图嵌入和动态图嵌入的模型策略。基于矩阵分解的方法只有包含特定的目标函数,才能学习相应的图结构和信息。基于随机游走的方法可以通过改变参数来控制游走方式,还可以与其他模型叠加提升性能。基于 AE 和 GNN 的方法利用近似定理^[130]广泛建模,使模型能够同时学习节点属性、拓扑结构和节点标签等信息。

图嵌入模型之间不是相互割裂的,而是存在理论依托关系: CGE 修改 LE 的目标函数,进一步增强邻近节点相似性; DANE 离线模型采用类似 LE 的方式保持各时刻快照的一阶相似度; DHPE 以 HOPE 为基础,引入矩阵摄动理论更新动态信息; NGE、MF 和 DWSF 以 NMF 为基础; node2vec 在 Deepwalk 的基础上引入有偏的随机游走; HARP 通常与 Deepwalk 或 node2vec 结合使用; dynnode2vec 在 node2vec 的基础上,使用 Skip-Gram 更新动态信

息; DNE-APP 在 DNGR 基础上引入成对约束; VGAE 使用 GCN 作为编码器; BurstGraph 使用 GraphSAGE 进行采样; GAT 在 GCN 的基础上引入注意力机制; MF-GCN 以 GraphSAGE 为基础构建; GraphAIR 将 GCN 和 GAT 作为组件; EvolveGCN 使用 RNN 调整 GCN 参数; TemporalGAT 对 GAT 和 TCN 进行集成。

3 数据集与应用

本节将详细介绍常见图嵌入数据集和机器学习任务。表 4 和表 5 分别为常用静态图和动态图嵌入数据集的统计信息。

3.1 数据集

3.1.1 静态图嵌入数据集

20-Newsgroup^[63]: 由大约 20000 个新闻组文档构成的数据集。这些文档根据主题划分成 20 个组,每个文档表示为每个词的 TF-IDF 分数向量,构建余弦相似图。为了证明聚类算法的稳健性,分别从 3、6 和 9 个不同的新闻组构建了三个图,使用的缩写 NG 是 Newsgroup 的缩写。

Flickr^[131]: 由照片分享网站 Flickr 上的用户组成的网络。网络中的边指示用户之间的联系关系。标签指示用户的兴趣组(例如黑白色摄影)。

DBLP^[53]: 引文网络数据集,每个顶点表示一个作者,从一个作者到另一个作者的参考文献数量由这两个作者之间的边权重来记录。标签上标明了研究人员发表研究成果的 4 个领域。

YouTube^[132]: YouTube 视频分享网站用户之间的社交网络。标签代表了喜欢视频类型(例如动漫视频)的观众群体。

Wikipedia^[133]: 网页共现网络,节点表示网页,边表示网页之间的超链接。Wikipedia 数据集的 TF-IDF 矩阵是描述节点特征的文本信息,节点标签是网页的类别。

Table 2 Strategies of static graph embedding

表 2 静态图嵌入模型策略归纳

模型分类	模型	时间	模型策略
矩阵分解	LLE ^[24]	2000	构造邻域保持映射，最小化重建损失函数
	GF ^[13]	2013	分解邻接矩阵，利用向量内积捕捉边的存在
	GraRep ^[25]	2015	使用 SVD 分解 k 步对数转移概率矩阵
	HOPE ^[27]	2016	GSVD 分解相似度矩阵，L2 范数保持高阶相似度
	NGE ^[30]	2013	使用 NMF 将输入分解为系数矩阵和嵌入矩阵
	LE ^[32]	2001	保持相连节点在嵌入空间尽可能靠近
	CGE ^[33]	2011	修改 LE 损失函数，保持低权值节点对相似性
	SPE ^[34]	2009	利用核矩阵生成关系矩阵 \tilde{A} ，最小化 A 和 \tilde{A} 的差异
随机游走	Deepwalk ^[49]	2014	使用随机游走采样节点，Skip-Gram 最大化节点共现概率
	node2vec ^[50]	2016	在 Deepwalk 的基础上引入有偏的随机游走
	HARP ^[51]	2017	利用原始图生成保留全局结构的压缩图
	Walklets ^[52]	2016	改进 Deepwalk，捕获节点与社区的从属关系并建模
	TriDNR ^[54]	2016	最大化节点标签，节点邻域，节点内容的共现概率
自编码器	GraphEncoder ^[63]	2014	利用 L2 损失函数重构图相似度矩阵
	SDNE ^[64]	2016	使用有监督和无监督组件分别保持一阶和二阶相似度
	DNGR ^[65]	2016	随机冲浪捕获图结构，生成 PPMI 矩阵输入 SDAE
	DNE-APP ^[67]	2017	使用 PPMI 度量和 k 步转移矩阵构建相似度聚合矩阵
	VGAE ^[69]	2016	引入 VAE，使用 GCN 编码器，使用内积解码器
	GALA ^[73]	2020	编码器执行拉普拉斯平滑，解码器执行拉普拉斯锐化
	ANE ^[76]	2018	施加对抗性正则化避免流形断裂
图神经网络	GCN ^[70]	2016	利用谱卷积一阶近似提高层间传播效率
	GraphSAGE ^[85]	2017	采样和聚合节点的局部邻域特征训练聚合器函数
	GAT ^[95]	2017	在 GCN 的基础上引入 self-attention 和 multi-head attention
	GIN ^[98]	2018	利用 GNN 和 WL 图同构测试保留图结构信息
	MF-GCN ^[100]	2020	使用多个局部 GCN 滤波器提取节点特征
	GraphAIR ^[102]	2020	邻域聚合模块组合节点特征示，邻域交互模块通过乘法运算显示建模
	SDGNN ^[103]	2021	在 GNN 的基础引入地位理论和平衡理论
其他	LINE ^[116]	2015	分别优化一阶和二阶相似度，将嵌入向量进行拼接
	DNRE ^[117]	2018	直接使用 LSTM 聚合邻域信息重建节点嵌入
	Graphormer ^[119]	2021	在标准 Transformer 基础上，引入有中心性编码、空间编码和边编码

Cora、CiteSeer、Pubmed^[134]：标准的引文网络基准数据集，节点表示论文，边表示一篇论文对另一篇论文的引用。节点特征是论文的词袋表示，节点标签是论文的学术主题。

Yelp^[135]：本地商业评论和社交网站，用户可以提交对商家的评论，并与其他人交流。由于缺乏标签信息，该数据集常用于链接预测。

3.1.2 动态图嵌入数据集

Epinions^[136]：产品评论网站数据集，基于评论的词袋模型生成节点属性，以用户评论的主要类别作为类别标签。该数据集有 16 个时间戳。

Hep-th^[137]：高能物理理论会议研究人员的合作网络，原始数据集包含 1993 年 1 月至 2003 年 4 月期间高能物理理论会议的论文摘要。

Autonomous Systems(AS)^[138]：由边界网关协议日志构建的用户通信网络。该数据集包含从 1997

年 11 月 8 日到 2000 年 1 月 2 日的 733 条通信记录，信网络。该数据集包含 1999 年 1 月至 2002 年 7 月通常按照年份将这些记录划分为不同快照。期间公司员工之间的电子邮件。

Enron^[139]: Enron 公司员工之间的电子邮件通

Table 3 Strategies of dynamic graph embedding

表 3 动态图嵌入模型策略归纳

模型分类	模型	时间	模型策略
矩阵分解	DANE ^[36]	2017	拉普拉斯特征映射捕获 t 时刻的结构和属性信息，矩阵摄动理论更新动态信息
	DHPE ^[38]	2018	GSVD 分解各时刻 Katz 矩阵，矩阵摄动理论更新动态信息
	TRIP ^[40]	2015	利用图中三角形个数构建特征函数，将特征对矩阵映射成嵌入向量
	TIMERS ^[41]	2017	SVD 最大误差界重启，消除增量更新积累的误差
	DWSF ^[45]	2017	将图中有限监督信息合并为标签，并在每次迭代中更新
随机游走	CTDNE ^[56]	2018	按照时间顺序对边进行遍历
	dynnode2vec ^[57]	2018	node2vec 初始化快照，对变化点执行随机游走，利用 Skip-Gram 更新动态信息
	STWalk ^[59]	2017	捕捉规定时间窗内节点的变化
	tNodeEmbed ^[60]	2019	模仿句子嵌入作为节点嵌入，捕捉节点角色和边的动态变化
自编码器	DynGEM ^[78]	2018	提出 PropSize 动态调节神经元个数，同时引入 L1 和 L2 正则化
	dyngraph2vec ^[79]	2019	组合 AE 和 LSTM，构建不同的编码器和解码器组合
	NetWalk ^[81]	2018	同时最小化节点距离和自编码器重构误差
	BurstGraph ^[84]	2019	将动态演化分为一般演化和突发演化，RNN 捕捉各时刻的图结构
	HVGNN ^[92]	2021	采用基于 TGNN 的双曲 VGAE
图神经网络	DyRep ^[106]	2018	基于自我传播、外源驱动、局部嵌入传播更新节点表示
	DySAT ^[108]	2020	结构注意力提取邻域特征，时间注意力捕捉多个时刻的表示
	EvolveGCN ^[110]	2019	在每个时刻使用 RNN 调整 GCN 参数
	DGNN ^[111]	2020	使用时态信息增强 LSTM 作为更新框架
	TemporalGAT ^[112]	2020	集成 GAT 和 TCN，self-attention 应用于邻域，TCN 用于动态信息更新
其他	HTNE ^[120]	2018	Hawkes 捕捉过去时刻对当前时刻邻域的影响，Skip-Gram 更新动态信息
	DynamicTriad ^[122]	2018	通过闭合三元组和开放三元组模拟图的动态演化
	M ² DNE ^[125]	2019	微观动态描述结构的形成，宏观动态描述规模的演化，利用二者交互生成嵌入
	CAW ^[126]	2021	回溯多个时刻的相邻链接编码因果关系，根据特征位置计数编码相应节点标识

Table 4 Statistics of static graph datasets

表 4 静态图数据集统计信息

Dataset	20-NewsGroup	Flickr	DBLP	YouTube	Wikipedia	Cora	CiteSeer	Pubmed	Yelp
#Nodes	1720,3224,5141	80513	29199	1138499	2405	2708	3327	19717	6569
#Edges	Fully connected	5899882	133664	2990443	17981	5429	4732	44338	95361
#Features	—	—	—	—	4973	1433	3703	500	—
#Labels	3,6,9	195	4	47	17	7	6	3	—

Table 5 Statistics of dynamic graph datasets

表 5 动态图数据集统计信息

Datasets	Epinions	Hep-th	AS	Enron	UCI
#Nodes	14180	1424-7980	7716	184	1809
#Edges	227642	2556-21036	10695-26467	63-591	16822
#Features	9936	—	—	—	—
#Labels	20	—	—	—	—
#Time Steps	16	60	100	128	13

UCI^[140]：加州大学在线学生社区用户之间的通信网络。节点表示用户，用户之间的消息通信表示边缘。每条边携带的时间指示用户何时通信。

3.2 网络重构

网络重构任务是利用学习到节点低维向量表示重新构建原始图的拓扑结构，评估生成的嵌入保持图结构信息的能力。通过计算基于节点嵌入的内积或者相似性来预测节点简是否存在链接，使用前 k 对顶点真实链接所占原始图中链接的比例来评估模型的重构表现。

网络重构任务通常分为三个步骤：（1）使用图嵌入模型生成嵌入表示。（2）计算节点对应的重构邻近度并进行排序。（3）计算前 k 对节点的真实链接的比例。

网络重构任务通常采用 MAP(Mean Average Precision)^[141]作为评价指标：

$$AP(i) = \frac{\sum_j Pr @ j(i) \cdot \Delta_i(j)}{|\{\Delta_i(j) = 1\}|} \quad (20)$$

$$MAP = \frac{\sum_i AP(i)}{|V|}$$

式中， $Pr @ j(i)$ 是节点 v_i 的精度， $\Delta_i(j) = 1$ 表示节点 i 和 j 之间存在连接。

好的网络嵌入方法能够捕捉到具有网络结构信息的嵌入表示，从而能够很好地重构网络。在 Yelp 数据集上，ANE 和 LINE 的结果要好于仅利用一阶相似度的 LE 和使用随机游走的 Deepwalk。可见，表示局部结构的一阶相似性和表示全局结构的二阶相似性在保持网络结构方面都起着重要的作用。由于 ANE 采用了对抗性训练，其表现略优于 LINE。各模型网络重构性能见图 24。

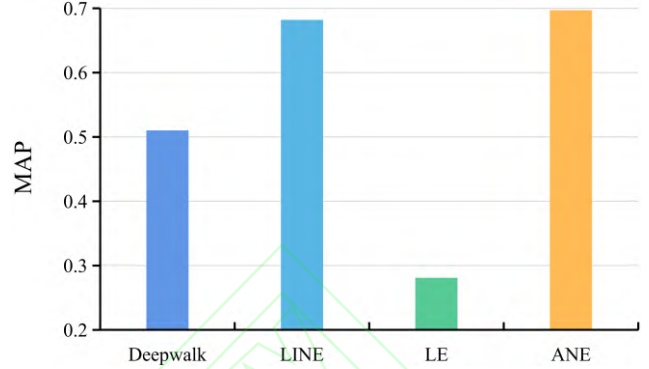


Fig.24 Comparisons of graph reconstruction

图 24 网络重构性能对比

3.3 节点分类

节点分类任务是利用图的拓扑结构和节点特征确定每个节点所属类别。现实世界的图数据，可能不是完全标签化的，由于各种因素大部分节点的标签可能是未知的，节点分类任务可以利用已有标签节点和连接关系来推断丢失的标签。此外，节点分类任务可分为两类：多标签分类和多类分类。前者使用的数据集中每个节点仅由一个类别标签标记，后者则由多个类别标签标记。

节点分类任务通常分为三个步骤：（1）使用图嵌入模型生成嵌入表示。（2）将包含标签信息的数据集划分为训练集和测试集。（3）在训练集上训练分类器，在测试集验证模型性能。常用的分类器包括逻辑回归分类器^[142]、最近邻分类器^[143]、支持向量机^[144]和朴素贝叶斯分类器^[145]等。

节点分类任务通常采用 *micro-F1* 和 *macro-F1* 作为评价指标：

$$macro-F1 = \frac{\sum_{l \in L} F1(l)}{|L|} \quad (21)$$

$$micro-F1 = \frac{2 * P * R}{P + R}$$

式中， $F1(l)$ 是标签 l 的 F1 分数， P 表示准确率， R 表示召回率。对于多分类任务准确度(Accuracy)与 *micro-F1* 的值相同。

利用网络结构和节点特征预测节点标签在网络分析中有着广泛的应用。通过将生成的嵌入作为节点特征对节点进行分类，可以比较各种嵌入方法

在该任务中的有效性。

在 CiteSeer 数据集上的静态图节点分类实验中,同时使用节点特征和网络拓扑结构的 GNN 模型均取得了较好的结果,仅使用网络结构进行无监督学习的 Deepwalk、node2vec 和 LINE 模型性能较差。相较原始的 GraphSAGE 模型, MF-GCN 使用多个局部 GCN 滤波器进行特征提取,使模型捕捉到节点特征的不同方面,从而进一步提升了模型性能。同样突出的 GraphAIR 模型分别对邻域聚合和邻域交互进行建模,与原始的 GCN 和 GAT 模型结合使用,提升基础模型的性能。各模型静态图节点分类性能见图 25。

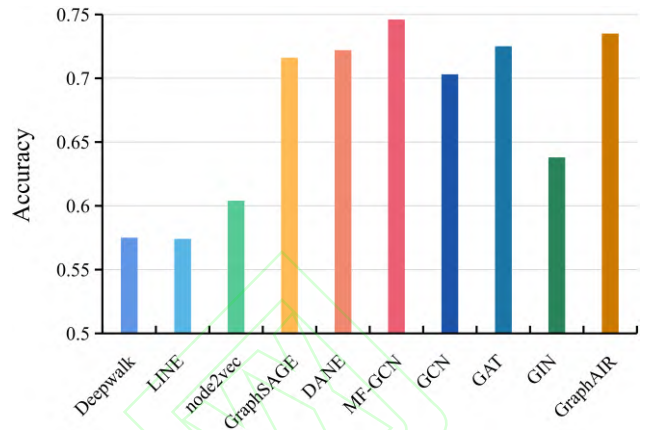


Fig.25 Comparisons of static graph node classification

图 25 静态图节点分类性能对比

Table 6 Comparisons of dynamic graph node classification

表 6 动态图节点分类性能对比

Models	Deepwalk	node2vec	LINE	DANE	tNodeEmbed	CTDNE	HTNE	DynamicTriad	M ² DNE
micro-F1	69.65	75.2	67.56	74.53	82.2	71.7	71.4	71.1	69.75
macro-F1	72.37	23.5	70.97	75.69	50.4	8.3	6.9	5.5	69.71

Table 7 Comparisons of link prediction

表 7 链接预测性能对比

Models	Deepwalk	GAE	VGAE	Linear-GAE	Linear-VGAE	GALA	GraphSAGE	MF-GCN	GraphAIR
AUC	80.5	89.5	90.8	91.5	91.6	94.4	93.7	92.4	95.0
AP	83.6	89.9	92.0	92.9	93.1	94.8	—	—	—

在引入时间信息的 DBLP 数据集上的动态图节点分类实验中,与静态图嵌入模型(DeepWalk、node2vec、LINE 和 SDNE)相比, DANE 保留的动态信息使生成的嵌入具有更强的表示能力;与动态图嵌入模型(tNodeEmbed、CTDNE、HTNE、DynamicTriad 和 M²DNE)相比, DANE 捕捉了拓扑结构和节点属性特征的相关性,并保留图的动态信息,生成抗噪声的嵌入,从而提高结构嵌入的准确性。此外,由于 DANE 抗噪声的特性,使其有较好的鲁棒性。各模型动态图节点分类性能见表 6。

3.4 链接预测

链接预测任务用于预测两个节点之间是否存在边或者预测图中未观察到的链接,评估生成嵌入在保持拓扑结构方面的性能。

链接预测任务通常分为三个步骤:(1)使用图嵌入模型生成嵌入表示。(2)对数据集中任意

两个节点间的边信息进行标记,然后将数据集分为训练集和测试集。(3)在训练集上训练分类器,在测试集上进行链接预测实验。

链接预测任务通常采用 AUC(Area Under the Curve)和 AP(Average Precision)作为评价指标:

- AUC: 把阈值设置在紧靠每个正例之下,计算负类的查全率,再取平均值。
- AP: 把阈值设置在紧靠每个正例之下,计算正类的查准率,再取平均值。

图嵌入可以显式或隐式地捕获网络的固有结构,以预测可能存在但未被观察到的连接关系。在 CiteSeer 数据集上, GraphAIR(AIR-GAE)的性能优于其他基于 GNN 和 AE 的方法, Deepwalk 的表现最差。主要原因可能是链路预测任务通常使用成对解码器来计算两个节点之间链路的概率。例如,

GAE 和 VGAE 假设两个节点之间存在边的概率与这两个节点的嵌入的点积成正比。AIR-GAE 通过两个节点嵌入相乘来显式地对邻域交互进行建模，这与链接预测任务有着内在的联系，进而提升了模型性能。各模型链接预测性能见表 7。

3.5 聚类

聚类任务采用无监督的方式将图划分为若干个社区，使属于同一社区的节点具有更多相似特性。在利用模型生成嵌入后，一般采用频谱聚类（非归一化谱聚类^[146]和归一化谱聚类^[147]）和 k -means^[148]等经典方法将节点嵌入聚类。

聚类任务通常采用归一化互信息(Normalized Mutual Information, NMI)^[149]评估聚类性能：

$$NMI = 1 - \frac{H(M_1 | M_2)_{norm} + H(M_2 | M_1)_{norm}}{2} \quad (22)$$

式中，NMI 用于度量 M_1 和 M_2 聚类结果之间的相似性， $H(\cdot)$ 表示信息熵。

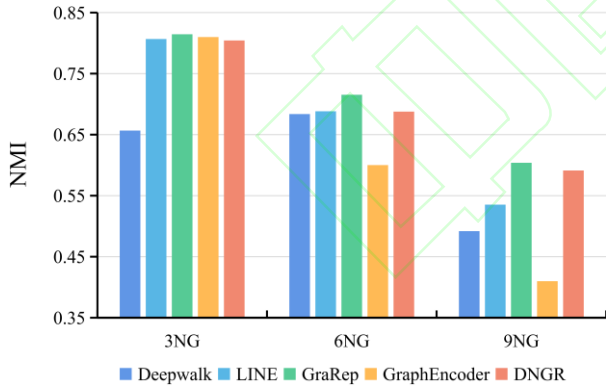


Fig.26 Comparisons of node clustering

图 26 聚类性能对比

将生成的嵌入表示进行聚类，使具有相似特性的节点尽可能在同一社区。在 20-NewsGroup 数据集上，GraRep、LINE 和 DNGR 模型在 3NG、6NG 和 9NG 实验中性能明显优于其他基线算法。对于 GraRep 和 LINE，两种方法可以有效捕捉丰富的局部关系信息，从而提高了聚类结果。对于 DNGR，使用的深度神经网络能够有效捕捉图的非线性，同时使用随机冲浪代替广泛使用的抽样方法加强了

原始图中信息的提取。各模型聚类性能见图 26。

3.6 异常检测

异常检测任务用于识别图中的“非正常”结构，通常包括异常节点检测、异常边缘检测和异常变化检测。常见的异常检测方法有两种：一是将原始图进行压缩，通过聚类和离群点检测识别压缩图中的异常^[150,151]；二是利用模型生成节点嵌入并分组为 k 个社群，检测不属于已有社群的新节点或边。

异常检测任务通常采用 AUC 作为评价指标。

图数据的异常检测主要是通过找出与正常数据集差异较大的离群点（异常点）。好的嵌入表示能够利用决策边界，有效界定正常点与异常点。在 DBLP 和 Hep-th 数据集上，Deepwalk、node2vec 和 SDNE 的实验结果相近，而 NetWalk 明显优于上述方法。NetWalk 模型的高性能得益于以下优点：（1）网络嵌入可以动态更新；（2）流式节点和边可以在存储空间恒定的情况下进行高效编码；（3）可以灵活的扩展到不同类型的网络；（4）可实时检测网络异常。各模型异常检测性能见图 27。

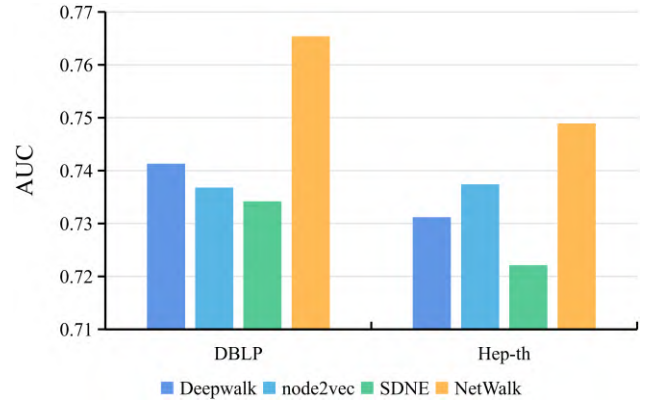


Fig.27 Comparisons of anomaly detection

图 27 异常检测性能对比

3.7 可视化

可视化任务是对嵌入进行降维和可视化，从而直观地观察原始图中某些特征。可视化任务通常是在有标签数据集上进行的，不同标签的节点在二维空间中使用不同的颜色进行标记。由于嵌入中保留了原始图的某种信息，所以可视化结果直接反映了二维空间中同一社群节点具有更加相似的特征。

对于可视化任务, 好的嵌入表示在二维图像中相同或相近的节点彼此接近, 不同的节点彼此分离。在 Cora 数据集上, 将模型生成的低维嵌入输入到 t -SNE^[10,152], 使嵌入维数降至 2, 同一类别的节点使用相同的颜色表示。Yuan 等人^[153]比较了不同模型的可视化性能, 部分可视化结果见图 28。GCN 和 GAT 学习的节点向量能够有效捕捉到社区的结构, 使同类型节点更为接近。Deepwalk 和 SDNE 只使用邻接矩阵作为输入, 没有充分利用节点特征和标签信息, 导致模型性能较差, 尤其是 SDNE 模型, 不同类型的点在二维空间中几乎是无序的。

4 未来研究方向

通过对传统和新型图嵌入方法的研究和分析, 现阶段的主要任务依然是提升模型对大规模和复杂图数据的扩展性、创新模型方法和提高下游任务性能。据此, 本节提出了四个未来工作的研究方向。

4.1 面向大规模图数据的嵌入模型

对于图嵌入任务, GF、LINE 和 GraphSAGE 等已经提出了大规模静态图分析的思路, 利用分布式计算或无监督学习的方式提高模型的计算效率。然

而, 对于具有复杂演化信息的大规模动态图, 已有动态图模型通常无法采用上述方法完成图表示学习任务。由于动态图通常表示为一组快照或带有时间戳的连续网络, 所以快照或时间戳的数量越多, 动态图的演化信息就会越复杂。因此, 降低网络演化复杂度和改进嵌入模型性能是解决大规模图嵌入问题的两个重要方面。

4.2 面向复杂图数据的嵌入模型

现阶段, 大部分研究仍集中在简单的同质图上, 如经典的 GCN 模型只需邻接矩阵、节点特征矩阵和系数矩阵即可实现。然而, 现实世界的网络往往更加复杂, 如异质图、有向图和超图等。现有模型中, HGSL^[154]实现了异质图表示学习, SDGNN 实现了符号有向图表示学习, DANE 实现了属性图的表示学习。随着图表示学习研究的深入, 探索更为复杂图数据的表示将仍旧是有前景的研究方向。

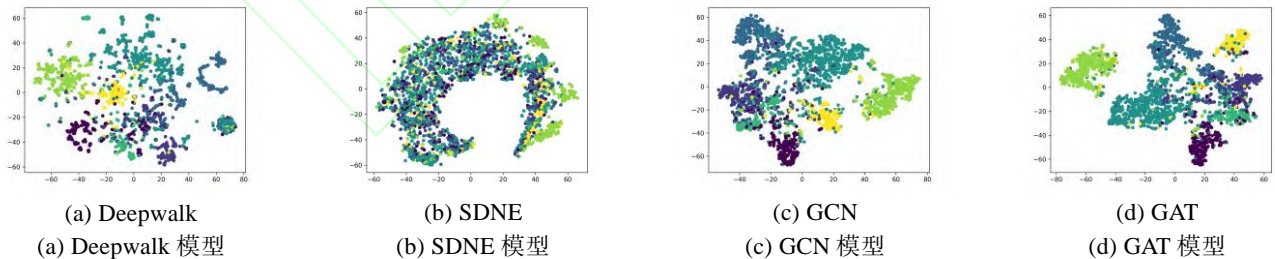


Fig.28 Visualization of different methods^[153]

图 28 不同模型的可视化结果^[153]

4.3 改进模型训练策略

复杂的模型可以提升效果, 但往往超参数较多且难以训练。相较于设计新的模型架构, 一些研究者开始探索如何利用训练策略(如数据扩增^[155])来提升现有模型的性能。GraphMix^[156]利用数据扩增技术, 将简单的 GCN 架构提升到先进基线模型的水平, 并且无需额外的内存和计算消耗。对于现阶段,

除了开发新的图嵌入模型外, 充分挖掘已有模型潜力仍然是一项充满挑战的工作。

4.4 面向特定任务的嵌入模型

大部分图嵌入模型生成的结果将用于节点分类、链接预测、可视化等多个任务。与上述建模思路不同, 面向任务的嵌入模型只关注一个任务, 并充分利用与该任务相关信息来训练模型。多数情况

下, 面向任务的嵌入模型在目标任务上比通用嵌入模型更有效, 如 Semi-AttentionAE^[153]采用集成学习的方法, 联合训练 GAT 与 LE, 提升节点分类任务性能。针对特定任务设计高性能模型同样是今后研究的热点方向。

5 结束语

本文对图嵌入文献进行了全面回顾, 给出了图嵌入有关定义, 提出了静态图和动态图模型通用的分类方法, 对现有模型的核心策略以及静态图和动态图模型理论相关性进行了系统性分析。在应用部分, 介绍了图嵌入常用数据集以及节点分类、链接预测等常见的机器学习任务, 比较了不同模型的表现。最后, 从图数据、模型策略和应用场景三个方面提出了图嵌入领域四个未来有希望的研究方向。

参考文献:

- [1] SONG Y M, GU Y, LI F F, et al. Survey on AI Powered New Techniques for Query Processing and Optimization[J]. Journal of Frontiers of Computer Science and Technology, 2020, 14(7): 1081-1103.
- [2] TRONCOSO F, WEBER R. A novel approach to detect associations in criminal networks[J]. Decision Support Systems, 2020, 128: 113159.
- [3] GUO S, LIN Y, FENG N, et al. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33(01): 922-929.
- [4] GU J, WANG Z, KUEN J, et al. Recent advances in convolutional neural networks[J]. Pattern Recognition, 2018, 77: 354-377.
- [5] SALEHINEJAD H, SANKAR S, BARFETT J, et al. Recent advances in recurrent neural networks[J]. arXiv preprint arXiv:1801.01078, 2017.
- [6] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. nature, 2015, 521(7553): 436-444.
- [7] BHAGAT S, CORMODE G, MUTHUKRISHNAN S. Node classification in social networks[M]//Social network data analytics. Springer, 2011: 115-148.
- [8] LIBEN-NOWELL D, KLEINBERG J. The link-prediction problem for social networks[J]. Journal of the American society for information science and technology, 2007, 58(7): 1019-1031.
- [9] DING C H, HE X, ZHA H, et al. A min-max cut algorithm for graph partitioning and data clustering[C]//Proceedings 2001 IEEE international conference on data mining. IEEE, 2001: 107-114.
- [10] VAN DER MAATEN L, HINTON G. Visualizing data using t-SNE[J]. Journal of machine learning research, 2008, 9(11).
- [11] QIU J, TANG J, MA H, et al. Deepinf: Modeling influence locality in large social networks[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD' 18). 2018.
- [12] SILVEIRA T, ZHANG M, LIN X, et al. How good your recommender system is? A survey on evaluations in recommendation[J]. International Journal of Machine Learning and Cybernetics, 2019, 10(5): 813-831.
- [13] AHMED A, SHERVASHIDZE N, NARAYANAMURTHY S, et al. Distributed large-scale natural graph factorization [C]//Proceedings of the 22nd international conference on World Wide Web. 2013: 37-48.
- [14] LOVÁSZ L. Random walks on graphs: A survey[J]. Combinatorics, Paul erdos is eighty, 1993, 2(1): 1-46.
- [15] GOLDBERG Y, LEVY O. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method[J]. arXiv preprint arXiv:1402.3722, 2014.
- [16] CAI H, ZHENG V W, CHANG K C-C. A comprehensive survey of graph embedding: Problems, techniques, and applications[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(9): 1616-1637.
- [17] GOYAL P, FERRARA E. Graph embedding techniques, applications, and performance: A survey[J]. Knowledge-Based Systems, 2018, 151: 78-94.
- [18] CHEN F, WANG Y-C, WANG B, et al. Graph representation learning: a survey[J]. APSIPA Transactions on Signal and Information Processing, 2020, 9.
- [19] CUI P, WANG X, PEI J, et al. A survey on network embedding[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 31(5): 833-852.
- [20] XIE Y, LI C, YU B, et al. A survey on dynamic network embedding[J]. arXiv preprint arXiv:2006.08093, 2020.
- [21] SKARDINGA J, GABRYS B, MUSIAL K. Foundations and modelling of dynamic networks using dynamic graph neural networks: A survey[J]. IEEE Access, 2021.
- [22] NG A. Sparse autoencoder[J]. CS294A Lecture notes, 2011, 72(2011): 1-19.
- [23] SCARSELLI F, GORI M, TSOI A C, et al. The graph neural network model[J]. IEEE transactions on neural networks, 2008, 20(1): 61-80.
- [24] ROWEIS S T, SAUL L K. Nonlinear dimensionality reduction by locally linear embedding[J]. science, 2000, 290(5500): 2323-2326.
- [25] CAO S, LU W, XU Q. Grarep: Learning graph representations with global structural information[C]//Proceedings of the 24th ACM international on conference on information and knowledge management. 2015: 891-900.

- [26] LEVY O, GOLDBERG Y. Neural word embedding as implicit matrix factorization[J]. *Advances in neural information processing systems*, 2014, 27: 2177–2185.
- [27] 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*. 2016: 1105–1114.
- [28] KATZ L. A new status index derived from sociometric analysis[J]. *Psychometrika*, 1953, 18(1): 39–43.
- [29] PAIGE C C, SAUNDERS M A. Towards a generalized singular value decomposition[J]. *SIAM Journal on Numerical Analysis*, 1981, 18(3): 398–405.
- [30] YANG J, YANG S, FU Y, et al. Non-negative graph embedding[C]//*2008 IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE, 2008: 1–8.
- [31] LEE D D, SEUNG H S. Learning the parts of objects by non-negative matrix factorization[J]. *Nature*, 1999, 401(6755): 788–791.
- [32] BELKIN M, NIYOGI P. Laplacian eigenmaps and spectral techniques for embedding and clustering.[C]//*Nips*. 2001, 14(14): 585–591.
- [33] LUO D, DING C H, NIE F, et al. Cauchy graph embedding[A]. *ICML[C]*. 2011.
- [34] SHAW B, JEBARA T. Structure preserving embedding[C]//*Proceedings of the 26th Annual International Conference on Machine Learning*. 2009: 937–944.
- [35] LI R-C. Matrix Perturbation Theory[G]//HOGBEN L. *Handbook of Linear Algebra*, Second Edition. New York, NY: Chapman and Hall/CRC, 2013, 20134740: 343–362.
- [36] 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*. 2017: 387–396.
- [37] 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.
- [38] ZHU D, CUI P, ZHANG Z, et al. High-order proximity preserved embedding for dynamic networks[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2018, 30(11): 2134–2144.
- [39] STRANG G, STRANG G, STRANG G, et al. *Introduction to linear algebra[M]*. Wellesley-Cambridge Press Wellesley, MA, 1993.
- [40] CHEN C, TONG H. Fast eigen-functions tracking on dynamic graphs[C]//*Proceedings of the 2015 SIAM international conference on data mining*. Philadelphia, PA: SIAM, 2015: 559–567.
- [41] TSOURAKAKIS C E. Fast counting of triangles in large real networks without counting: Algorithms and laws[C]//*2008 Eighth IEEE International Conference on Data Mining*. Piscataway, NJ: IEEE, 2008: 608–617.
- [42] TSOURAKAKIS C E. Counting triangles in real-world networks using projections[J]. *Knowledge and Information Systems*, 2011, 26(3): 501–520.
- [43] ZHANG Z, CUI P, PEI J, et al. Timers: Error-bounded svd restart on dynamic networks[C]//*Proceedings of the AAAI Conference on Artificial Intelligence*. 2018, 32(1).
- [44] AHMED N M, CHEN L, WANG Y, et al. DeepEye: link prediction in dynamic networks based on non-negative matrix factorization[J]. *Big Data Mining and Analytics*, 2018, 1(1): 19–33.
- [45] SEYEDI S A, MORADI P, TAB F A. A weakly-supervised factorization method with dynamic graph embedding[C]//*2017 Artificial Intelligence and Signal Processing Conference (AISP)*. Piscataway, NJ: IEEE, 2017: 213–218.
- [46] WANG Y-X, ZHANG Y-J. Nonnegative matrix factorization: A comprehensive review[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2012, 25(6): 1336–1353.
- [47] TRIGEORGIS G, BOUSMALIS K, ZAFEIRIOU S, et al. A deep matrix factorization method for learning attribute representations[J]. *IEEE transactions on pattern analysis and machine intelligence*, 2016, 39(3): 417–429.
- [48] LIU Y, LEE J, PARK M, et al. Learning to Propagate Labels: Transductive Propagation Network for Few-shot Learning[J]. *arXiv:1805.10002 [cs, stat]*, 2019.
- [49] 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*. 2014: 701–710.
- [50] 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*. 2016: 855–864.
- [51] CHEN H, PEROZZI B, HU Y, et al. Harp: Hierarchical representation learning for networks[C]//*Proceedings of the AAAI Conference on Artificial Intelligence*. 2018, 32(1).
- [52] YANG C, LIU Z. Comprehend deepwalk as matrix factorization[J]. *arXiv preprint arXiv:1501.00358*, 2015.
- [53] PEROZZI B, KULKARNI V, SKIENA S. Walklets: Multiscale graph embeddings for interpretable network classification[J]. *arXiv preprint arXiv:1605.02115*, 2016: 043238–23.
- [54] PAN S, WU J, ZHU X, et al. Tri-party deep network representation[J]. *Network*, 2016, 11(9): 12.
- [55] SAJJAD H P, DOCHERTY A, TYSHETSKIY Y. Efficient representation learning using random walks for dynamic graphs[J]. *arXiv preprint arXiv:1901.01346*, 2019.
- [56] 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*. 2018: 969–976.
- [57] MAHDAVI S, KHOSHRAFTAR S, AN A. dynnode2vec:

- Scalable dynamic network embedding[C]//2018 IEEE International Conference on Big Data (Big Data). Piscataway, NJ: IEEE, 2018: 3762–3765.
- [58] KIM Y, CHIU Y-I, HANAKI K, et al. Temporal analysis of language through neural language models[J]. arXiv preprint arXiv:1405.3515, 2014.
- [59] PANDHRE S, MITTAL H, GUPTA M, et al. Stwalk: learning trajectory representations in temporal graphs[C]//Proceedings of the ACM India joint international conference on data science and management of data. 2018: 210–219.
- [60] SINGER U, GUY I, RADINSKY K. Node embedding over temporal graphs[J]. arXiv preprint arXiv:1903.08889, 2019.
- [61] PALANGI H, PALANGI H, DENG L, et al. Deep Sentence Embedding Using the Long Short Term Memory Network: Analysis and Application to Information Retrieval. arXiv.org[M]. February, 2015.
- [62] BOURLARD H, KAMP Y. Auto-association by multilayer perceptrons and singular value decomposition[J]. Biological cybernetics, 1988, 59(4): 291–294.
- [63] TIAN F, GAO B, CUI Q, et al. Learning deep representations for graph clustering[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2014, 28(1).
- [64] 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. 2016: 1225–1234.
- [65] CAO S, LU W, XU Q. Deep neural networks for learning graph representations[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2016, 30(1).
- [66] BULLINARIA J A, LEVY J P. Extracting semantic representations from word co-occurrence statistics: A computational study[J]. Behavior research methods, 2007, 39(3): 510–526.
- [67] SHEN X, CHUNG F-L. Deep network embedding with aggregated proximity preserving[C]//Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017. 2017: 40–43.
- [68] KINGMA D P, WELING M. Auto-encoding variational bayes[J]. arXiv preprint arXiv:1312.6114, 2013.
- [69] KIPF T N, WELING M. Variational graph auto-encoders[J]. arXiv preprint arXiv:1611.07308, 2016.
- [70] KIPF T N, WELING M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.
- [71] KULLBACK S, LEIBLER R A. On information and sufficiency[J]. The annals of mathematical statistics, 1951, 22(1): 79–86.
- [72] SALHA G, HENNEQUIN R, VAZIRGIANNIS M. Keep it simple: Graph autoencoders without graph convolutional networks[J]. arXiv preprint arXiv:1910.00942, 2019.
- [73] PARK J, LEE M, CHANG H J, et al. Symmetric graph convolutional autoencoder for unsupervised graph representation learning[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 6519–6528.
- [74] CAI D, HE X, HU Y, et al. Learning a spatially smooth subspace for face recognition[C]//2007 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2007: 1–7.
- [75] TAUBIN G. A signal processing approach to fair surface design[C]//Proceedings of the 22nd annual conference on Computer graphics and interactive techniques. 1995: 351–358.
- [76] XIAO Y, XIAO D, HU B, et al. ANE: Network embedding via adversarial autoencoders[C]//2018 IEEE International Conference on Big Data and Smart Computing (BigComp). Piscataway, NJ: IEEE, 2018: 66–73.
- [77] MAKHZANI A, SHLENS J, JAITLEY N, et al. Adversarial autoencoders[J]. arXiv preprint arXiv:1511.05644, 2015.
- [78] GOYAL P, KAMRA N, HE X, et al. Dyngem: Deep embedding method for dynamic graphs[J]. arXiv preprint arXiv:1805.11273, 2018.
- [79] GOYAL P, CHHETRI S R, CANEDO A. dyngraph2vec: Capturing network dynamics using dynamic graph representation learning[J]. Knowledge-Based Systems, 2020, 187: 104816.
- [80] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735–1780.
- [81] YU W, CHENG W, AGGARWAL C C, et al. Netwalk: A flexible deep embedding approach for anomaly detection in dynamic networks[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018: 2672–2681.
- [82] AKOGLU L, FALOUTSOS C. Anomaly, event, and fraud detection in large network datasets[C]//Proceedings of the sixth ACM international conference on Web search and data mining. 2013: 773–774.
- [83] AKOGLU L, TONG H, KOUTRA D. Graph based anomaly detection and description: a survey[J]. Data mining and knowledge discovery, 2015, 29(3): 626–688.
- [84] ZHAO Y, WANG X, YANG H, et al. Large Scale Evolving Graphs with Burst Detection[C]//IJCAI. 2019: 4412–4418.
- [85] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs[J]. arXiv preprint arXiv:1706.02216, 2017.
- [86] MITCHELL T J, BEAUCHAMP J J. Bayesian variable selection in linear regression[J]. Journal of the american statistical association, 1988, 83(404): 1023–1032.
- [87] HEARD N A, WESTON D J, PLATANIOTI K, et al. Bayesian anomaly detection methods for social networks[J]. The Annals of Applied Statistics, 2010: 645–662.
- [88] KLEINBERG J. Bursty and hierarchical structure in

- streams[J]. *Data mining and knowledge discovery*, 2003, 7(4): 373–397.
- [89] CHEN W, FANG W, HU G, et al. On the hyperbolicity of small-world and treelike random graphs[J]. *Internet Mathematics*, 2013, 9(4): 434–491.
- [90] GANEA O-E, BÉCIGNEUL G, HOFMANN T. Hyperbolic neural networks[J]. *arXiv preprint arXiv:1805.09112*, 2018.
- [91] 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*. 2018: 2827–2836.
- [92] SUN L, ZHANG Z, ZHANG J, et al. Hyperbolic Variational Graph Neural Network for Modeling Dynamic Graphs[C]//*Proceedings of the AAAI Conference on Artificial Intelligence*. 2021, 35(5): 4375–4383.
- [93] HAMMOND D K, VANDERGHEYNST P, GRIBONVAL R. Wavelets on graphs via spectral graph theory[J]. *Applied and Computational Harmonic Analysis*, 2011, 30(2): 129–150.
- [94] DEFFERRARD M, BRESSON X, VANDERGHEYNST P. Convolutional neural networks on graphs with fast localized spectral filtering[J]. *arXiv preprint arXiv:1606.09375*, 2016.
- [95] VELIČKOVIĆ P, CUCURULL G, CASANOVA A, et al. Graph attention networks[J]. *arXiv preprint arXiv:1710.10903*, 2017.
- [96] CHAUDHARI S, MITHAL V, POLATKAN G, et al. An attentive survey of attention models[J]. *arXiv preprint arXiv:1904.02874*, 2019.
- [97] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. *arXiv preprint arXiv:1706.03762*, 2017.
- [98] XU K, HU W, LESKOVEC J, et al. How powerful are graph neural networks?[J]. *arXiv preprint arXiv:1810.00826*, 2018.
- [99] LEMAN A A, WEISFEILER B. A reduction of a graph to a canonical form and an algebra arising during this reduction[J]. *Nauchno-Tekhnicheskaya Informatsiya*, 1968, 2(9): 12–16.
- [100] WANYAN T, ZHANG C, AZAD A, et al. Attribute2vec: Deep Network Embedding Through Multi-Filtering GCN[J]. *arXiv preprint arXiv:2004.01375*, 2020.
- [101] WU F, SOUZA A, ZHANG T, et al. Simplifying graph convolutional networks[C]//*International conference on machine learning*. Cambridge MA: PMLR, 2019: 6861–6871.
- [102] HU F, ZHU Y, WU S, et al. Graphair: Graph representation learning with neighborhood aggregation and interaction[J]. *Pattern Recognition*, 2021, 112: 107745.
- [103] HUANG J, SHEN H, HOU L, et al. SDGNN: Learning Node Representation for Signed Directed Networks[J]. *arXiv preprint arXiv:2101.02390*, 2021.
- [104] LESKOVEC J, HUTTENLOCHER D, KLEINBERG J. Predicting positive and negative links in online social networks[C]//*Proceedings of the 19th international conference on World wide web*. 2010: 641–650.
- [105] LESKOVEC J, HUTTENLOCHER D, KLEINBERG J. Signed networks in social media[C]//*Proceedings of the SIGCHI conference on human factors in computing systems*. 2010: 1361–1370.
- [106] TRIVEDI R, FARAJTABAR M, BISWAL P, et al. Dyrep: Learning representations over dynamic graphs[C]//*International Conference on Learning Representations*. 2019.
- [107] ZHANG J, SHI X, XIE J, et al. Gaan: Gated attention networks for learning on large and spatiotemporal graphs[J]. *arXiv preprint arXiv:1803.07294*, 2018.
- [108] SANKAR A, WU Y, GOU L, et al. Dysat: Deep neural representation learning on dynamic graphs via self-attention networks[C]//*Proceedings of the 13th International Conference on Web Search and Data Mining*. 2020: 519–527.
- [109] GEHRING J, AULI M, GRANGIER D, et al. Convolutional sequence to sequence learning[C]//*International Conference on Machine Learning*. Cambridge MA: PMLR, 2017: 1243–1252.
- [110] PAREJA A, DOMENICONI G, CHEN J, et al. Evolvegc: Evolving graph convolutional networks for dynamic graphs[C]//*Proceedings of the AAAI Conference on Artificial Intelligence*. 2020, 34(04): 5363–5370.
- [111] MA Y, GUO Z, REN Z, et al. Streaming graph neural networks[C]//*Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2020: 719–728.
- [112] FATHY A, LI K. TemporalGAT: Attention-Based Dynamic Graph Representation Learning[C]//*Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2020: 413–423.
- [113] BAI S, KOLTER J Z, KOLTUN V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling[J]. *arXiv preprint arXiv:1803.01271*, 2018.
- [114] OORD A van den, DIELEMAN S, ZEN H, et al. Wavenet: A generative model for raw audio[J]. *arXiv preprint arXiv:1609.03499*, 2016.
- [115] SANKAR A, WU Y, GOU L, et al. Dynamic graph representation learning via self-attention networks[J]. *arXiv preprint arXiv:1812.09430*, 2018.
- [116] 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*. 2015: 1067–1077.
- [117] 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. 2018: 2357–2366.
- [118] PAGE L, BRIN S, MOTWANI R, et al. The PageRank citation ranking: Bringing order to the web.[R]. Stanford InfoLab, 1999.
- [119] YING C, CAI T, LUO S, et al. Do Transformers Really Perform Bad for Graph Representation?[J]. arXiv preprint arXiv:2106.05234, 2021.
- [120] ZUO Y, LIU G, LIN H, et al. Embedding temporal network via neighborhood formation[C]//Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining. 2018: 2857–2866.
- [121] HAWKES A G. Spectral of some self-exciting and mutually exciting point processes[J]. Biometrika, Oxford University Press, 1971, 58(1): 83–90.
- [122] 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. 2018, 32(1).
- [123] COLEMAN J S. Foundations of social theory[M]. Harvard university press, 1994.
- [124] HUANG H, TANG J, LIU L, et al. Triadic closure pattern analysis and prediction in social networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(12): 3374–3389.
- [125] 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. 2019: 469–478.
- [126] WANG Y, CHANG Y-Y, LIU Y, et al. Inductive Representation Learning in Temporal Networks via Causal Anonymous Walks[J]. arXiv preprint arXiv:2101.05974, 2021.
- [127] MICALI S, ZHU Z A. Reconstructing markov processes from independent and anonymous experiments[J]. Discrete Applied Mathematics, 2016, 200: 108–122.
- [128] AHMED N K, NEVILLE J, ROSSI R A, et al. Efficient graphlet counting for large networks[C]//2015 IEEE International Conference on Data Mining. Piscataway, NJ: IEEE, 2015: 1–10.
- [129] PARANJAPE A, BENSON A R, LESKOVEC J. Motifs in temporal networks[C]//Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. 2017: 601–610.
- [130] HORNIK K, STINCHCOMBE M, WHITE H. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks[J]. Neural networks, 1990, 3(5): 551–560.
- [131] ZAFARANI R, LIU H. Social computing data repository at ASU[M]. 2009.
- [132] TANG L, LIU H. Scalable learning of collective behavior based on sparse social dimensions[C]//Proceedings of the 18th ACM conference on Information and knowledge management. 2009: 1107–1116.
- [133] YANG C, LIU Z, ZHAO D, et al. Network representation learning with rich text information[C]//Twenty-fourth international joint conference on artificial intelligence. 2015.
- [134] SEN P, NAMATA G, BILGIC M, et al. Collective classification in network data[J]. AI magazine, 2008, 29(3): 93–93.
- [135] ZENG H, ZHOU H, SRIVASTAVA A, et al. Graphsaint: Graph sampling based inductive learning method[J]. arXiv preprint arXiv:1907.04931, 2019.
- [136] TANG J, GAO H, LIU H. mTrust: Discerning multi-faceted trust in a connected world[C]//Proceedings of the fifth ACM international conference on Web search and data mining. 2012: 93–102.
- [137] GEHRKE J, GINSBURG P, KLEINBERG J. Overview of the 2003 KDD Cup[J]. Acm Sigkdd Explorations Newsletter, 2003, 5(2): 149–151.
- [138] LESKOVEC J, KREVL A. SNAP Datasets: Stanford large network dataset collection[M]. 2014.
- [139] KLIMT B, YANG Y. The enron corpus: A new dataset for email classification research[C]//European Conference on Machine Learning. Berlin, German: Springer, 2004: 217–226.
- [140] PANZARASA P, OPSAHL T, CARLEY K M. Patterns and dynamics of users' behavior and interaction: Network analysis of an online community[J]. Journal of the American Society for Information Science and Technology, 2009, 60(5): 911–932.
- [141] KUMARAN G, ALLAN J. Adapting information retrieval systems to user queries[J]. Information Processing & Management, 2008, 44(6): 1838–1862.
- [142] HOSMER D W, LEMESHOW S, STURDIVANT R X. Applied logistic regression[M]. Wiley New York, 2000.
- [143] PEDREGOSA F, VAROQUAUX G, GRAMFORT A, et al. Scikit-learn: Machine learning in Python[J]. the Journal of machine Learning research, 2011, 12: 2825–2830.
- [144] SUYKENS J A, VANDEWALLE J. Least squares support vector machine classifiers[J]. Neural processing letters, 1999, 9(3): 293–300.
- [145] MCCALLUM A, NIGAM K. A comparison of event models for naive bayes text classification[C]//AAAI-98 workshop on learning for text categorization. New York, NY: Citeseer, 1998: 41–48.
- [146] VON LUXBURG U. A tutorial on spectral clustering[J]. Statistics and computing, 2007, 17(4): 395–416.
- [147] SHI J, MALIK J. Normalized cuts and image segmentation[J]. IEEE Transactions on pattern analysis and machine intelligence, 2000, 22(8): 888–905.
- [148] MACQUEEN J. Some methods for classification and analysis of multivariate observations[C]//Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. 1967: 281–297.

- [149] CAI D, HE X, HAN J, et al. Graph regularized nonnegative matrix factorization for data representation[J]. IEEE transactions on pattern analysis and machine intelligence, 2010, 33(8): 1548–1560.
- [150] MANZOOR E, MILAJERDI S M, AKOGLU L. Fast memory-efficient anomaly detection in streaming heterogeneous graphs[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016: 1035–1044.
- [151] RANSHOUS S, HARENBERG S, SHARMA K, et al. A scalable approach for outlier detection in edge streams using sketch-based approximations[C]//Proceedings of the 2016 SIAM International Conference on Data Mining. Philadelphia, PA: SIAM, 2016: 189–197.
- [152] TANG J, LIU J, ZHANG M, et al. Visualizing large-scale and high-dimensional data[C]//Proceedings of the 25th international conference on world wide web. 2016: 287–297.
- [153] YUAN L, WANG Y, CHENG X, et al. Semi-AttentionAE: An Integrated Model for Graph Representation Learning[J]. IEEE Access, 2021, 9: 80787–80796.
- [154] ZHAO J, WANG X, SHI C, et al. Heterogeneous Graph Structure Learning for Graph Neural Networks[C]//35th AAAI Conference on Artificial Intelligence (AAAI). 2021.
- [155] VAN DYK D A, MENG X-L. The art of data augmentation[J]. Journal of Computational and Graphical Statistics, 2001, 10(1): 1–50.
- [156] VERMA V, QU M, KAWAGUCHI K, et al. GraphMix: Improved Training of GNNs for Semi-Supervised Learning[J]. arXiv e-prints, 2019: arXiv-1909.



袁立宁(1995-), 男, 硕士研究生, CCF 学生会员, 主要研究方向包括机器学习, 图神经网络等。

YUAN Lining, born in 1995, M.S. candidate, student member of CCF. His research interests include machine learning and graph neural network.



李欣(1987-), 男, 硕士, 天津市公安局河西分局科技信息化支队警务技术四级主管, 主要研究方向包括机器学习, 公安信息化等。

LI Xin, born in 1987, M.A. His research interests include machine learning and policing information system.



王晓冬(1980-), 男, 天津市公安局河东分局科信支队警务信息侦控工程师, 主要研究方向包括警务大数据, 技术侦查等。

WANG Xiaodong, born in 1980, intermediate engineer. His main research interests include police big data and technical investigation.



刘钊(1981-), 男, 博士, 中国人民公安大学讲师, 主要研究方向包括机器学习, 计算机视觉。

LIU Zhao, born in 1981, Ph.D, lecturer. His main research interests include machine learning and computer vision.