



# 计算机科学

COMPUTER SCIENCE

## 融合快速注意力机制的节点无特征网络链路预测算法

李勇, 吴京鹏, 张钟颖, 张强

引用本文

李勇, 吴京鹏, 张钟颖, 张强. [融合快速注意力机制的节点无特征网络链路预测算法](#)[J]. 计算机科学, 2022, 49(4): 43-48.

Li Yong, WU Jing-peng, ZHANG Zhong-ying, ZHANG Qiang. [Link Prediction for Node Featureless Networks Based on Faster Attention Mechanism](#)[J]. Computer Science, 2022, 49(4): 43-48.

---

## 相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

### [基于时空自适应图卷积神经网络的脑电信号情绪识别](#)

EEG Emotion Recognition Based on Spatiotemporal Self-Adaptive Graph Convolutional Neural Network

计算机科学, 2022, 49(4): 30-36. <https://doi.org/10.11896/jsjcx.210900200>

### [图神经网络在 Text-to-SQL 解析中的技术研究](#)

Technical Research of Graph Neural Network for Text-to-SQL Parsing

计算机科学, 2022, 49(4): 110-115. <https://doi.org/10.11896/jsjcx.210200173>

### [基于 BERT 的端到端语音合成方法](#)

End-to-End Speech Synthesis Based on BERT

计算机科学, 2022, 49(4): 221-226. <https://doi.org/10.11896/jsjcx.210300071>

### [改进 YOLOv3 网络模型的人体异常行为检测方法](#)

Human Abnormal Behavior Detection Method Based on Improved YOLOv3 Network Model

计算机科学, 2022, 49(4): 233-238. <https://doi.org/10.11896/jsjcx.210300251>

### [基于 Word2Vec 和改进注意力机制 AlexNet-2 的文本分类方法](#)

Text Classification Method Based on Word2Vec and AlexNet-2 with Improved Attention Mechanism

计算机科学, 2022, 49(4): 288-293. <https://doi.org/10.11896/jsjcx.211100016>

# 融合快速注意力机制的节点无特征网络链路预测算法

李勇 吴京鹏 张钟颖 张强

西北师范大学计算机科学与工程学院 兰州 730070

(facingworld@nwnu.edu.cn)

**摘要** 链路预测是网络科学的一个重要研究分支,旨在推断网络中节点对间存在连边的可能性。现实生活中很多事物关系都能够通过网络科学进行描述,很多实际问题都可以转化为链路预测问题。节点无特征网络链路预测算法可向有向网络、加权网络、时序网络等更复杂的网络推广。但现有的链路预测算法面临着网络结构信息挖掘不够深入、特征提取过程受人为主观意识影响、算法很难迁移到其他网络中、算法复杂度过高而无法在大型真实工业网络中应用等诸多问题。针对上述问题,文中基于图注意力网络的基本结构,采用图嵌入表示技术采集节点特征,类比神经图灵机中的内存寻址策略,结合复杂网络重要节点发现的相关工作,设计了一种快速高效的注意力计算方式,提出了一种融合快速注意力机制的节点无特征网络链路预测算法(Faster Attention Mechanism Link Prediction Algorithm, FALP)。在3个公开数据集和1个私有数据集上进行实验,结果表明, FALP算法有效避免了上述问题,同时具有优异的预测性能。

**关键词:** 链路预测; 注意力机制; 图神经网络; 图嵌入表示; 网络科学

中图法分类号 TP312

## Link Prediction for Node Featureless Networks Based on Faster Attention Mechanism

LI Yong, WU Jing-peng, ZHANG Zhong-ying and ZHANG Qiang

College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

**Abstract** Link prediction is an important task in network science. It aims to predict the link existence probabilities of two nodes. There are many relations between substances in real word, which can be described by network science in computers. There are many problems of daily life, which can be transformed to link prediction tasks. Link prediction algorithms for node featureless networks are convenient to migrate in directed networks, weighted networks, time networks, and so on. However, the traditional link prediction algorithms are faced with many problems as follows. The network structures information mining is not deep enough. The feature extraction processes depend on subjective consciousness. The algorithms are short of universality, and the time complexity and space complexity are flawed, which cause that they are difficult to be applied to real industry networks. In order to effectively avoid the above problems, based on the basic structure of graph attention network, graph embedding representation technology is used to collect node characteristics, analogy with the memory addressing strategy in neural turing machine, and combined with the relevant work of important node discovery in complex network, a fast and efficient attention calculation method is designed, and a node featureless network link prediction algorithm FALP integrating fast attention mechanism is proposed. Experiment on three public datasets and a private dataset show that the FALP effectively avoids these problems and has excellent predictive performance.

**Keywords** Link prediction, Attention mechanism, Graph neural networks (GNNs), Graph embedding, Network science

## 1 引言

在计算机中通常以实体的形式描述真实世界的事物,不同实体之间存在的关系以节点和边的形式构成了复杂网络,

如蛋白质-蛋白质相互作用网络<sup>[1]</sup>、论文引文网络<sup>[2]</sup>、分子网络<sup>[3]</sup>、交通网络<sup>[4]</sup>等。通过分析边的变化规律,可以了解网络的基本演化过程,从而映射出事物的本质及其潜在的发展规律。链路预测作为分析网络变化规律的一种重要工具,其

到稿日期:2021-08-31 返修日期:2021-12-12

基金项目:国家自然科学基金(72161034, 61863032);西北师范大学重大科研项目(NWNU-LKZD2021-06);全国高等院校计算机基础教育教学研究项目(2020-AFCEC-355);甘肃省教育科学规划课题研究项目(GS[2018]GHBBKZ021)

This work was supported by the National Natural Science Foundation of China (72161034, 61863032), Major Scientific Research Projects of Northwest Normal University (NWNU-LKZD2021-06), Research Project on Association of Fundamental Computing Education in Chinese Universities (2020-AFCEC-355) and Research Project on Educational Science Planning of Gansu (GS[2018]GHBBKZ021).

通信作者:吴京鹏(wu.jing.peng@qq.com)

主要目的在于量化网络中节点间存在连边的可能性。

在社会科学中,对病毒传播网络的链路进行预测可以有效避免传染病的大规模爆发<sup>[5]</sup>;对罪犯社会关系网络的链路进行预测可以提前发现潜在的社会威胁并加以制止<sup>[6]</sup>。在自然科学中,对分子网络的链路进行预测可以模拟化合物的合成过程,从而降低实验成本<sup>[7]</sup>;对电力网络的链路进行预测可以使供电网络的布局更加合理,有效避免级联输电故障的发生<sup>[8]</sup>。此外,在商品购买、在线社交等网络中,链路预测任务常被用来设计推荐系统<sup>[9]</sup>,具有较高的商业价值。

现有的链路预测算法可以分为以下3类:

(1)基于节点相似性的方法。该类方法首先计算出两个节点间的相似度;然后判断相似度的高低,相似度越高,两个节点间产生连边的可能性就越大,反之越小;最后拟合出一个相似度阈值,通过阈值判断两个节点间是否存在连边。该类算法的性能普遍较低,原因在于:1)基于相似度假设的策略在某些情况下适得其反,例如在分子网络中,原子结构、原子核外电子数差异较大,即节点间相似度较低时,构成稳定化合物的可能性反而越大;2)大多数节点的相似度算法仅考虑本地网络结构,在网络呈现出节点数量较多且连边数量较少的情况下,其计算出的相似度无法体现差异性。

(2)基于机器学习的方法。该类方法首先通过一定的规则来确定正例样本和反例样本;然后通过人为定义的特征选取策略为样本指定特征向量;最后通过二分类器给出两个节点间存在连边的概率。该类方法的性能较基于节点相似性方法有所提高,但仍不理想,原因在于:1)人为地为样本指定特征向量不具有客观性与全面性;2)人为指定的特征向量通常维度较低,不利于机器学习模型进行数据拟合。同时,该类方法的普适性较低,原因在于在不同的网络中节点对应的实体不同,而不同的实体往往具有不同含义的特征向量。

(3)基于图嵌入的方法。该方法通过随机游走、矩阵分解、图卷积神经网络等技术生成节点的向量表示。这种由程序生成的节点特征向量通过深度学习技术不断迭代学习,从而可表达丰富的网络结构信息。此外,这种生成的节点特征向量维数远高于人为指定的特征向量,更便于分类器进行数据拟合。然而,随机游走会导致算法性能不稳定,模型需要多次训练再从中择优;基于矩阵分解指对相似度矩阵或邻接矩阵进行分解,因此该类算法往往复杂度较高,而且在节点和边相对稀疏的网络中性能低下;基于图卷积神经网络算法设计出的链路预测模型具有十分优异的性能,但是该类算法的卷积操作往往需要耗费大量的硬件资源与时间,同时该类算法还要求节点具有较多的特征属性,而现实生活中的网络通常不具备这些条件。

本文的主要贡献如下:

(1)提出了一种快速高效的图注意力卷积策略,相比传统图注意力卷积策略,所提算法不仅避免了三维矩阵存储,而且将其时间复杂度由平方阶降至线性阶,同时还保留了较好的预测性能。

(2)将图随机游走与图卷积神经网络技术相融合,使得节点特征向量维数可变,将图注意力网络在节点无特征网络中

进行推广,提高了算法的普适性与灵活性。

(3)FALP算法适用于节点无特征网络的节点分类、社区检测等其他任务,具有较强的功能扩展性。

## 2 相关工作

基于节点相似度的链路预测算法认为,两个节点之间的相似度越高,两个节点间存在连边的可能性就越大。此类算法的差异主要体现在节点相似度的计算方式上,大致可分为两类:1)基于中心测度的节点相似性算法,其核心思想是通过1阶邻居网络的度、权值和共同邻居数量定义节点相似度,如CN(Common Neighbor),Jaccard等;2)基于径向测度的节点相似度算法,其核心思想是在节点的 $n$ 阶邻居网络中,按照一定规则选取部分节点的度、权值和共同邻居数量等信息定义节点相似度,如Kata,LHN2等。关于此类算法的详细介绍请参考文献<sup>[10]</sup>。

此类算法的不足之处在于:1)度、权值和邻居数量等基本网络结构参数表达的信息有限,尤其是在网络集聚系数较低的情况下,此类方法计算出的节点相似度将无法体现差异性;2)已有研究表明<sup>[11]</sup>,基于相似性的连边假设仅仅在大部分社交网络中有效,对于大部分非社交网络,其节点间的相似性越低,产生连边的可能性越大,例如,在邮件网络中,相互认识的人互相发送邮件的可能性较大;然而,在化合物网络中,元素的性质和结构差异较大时,构成稳定化合物的可能性越大。Zhang等也注意到上述问题,采用集成学习的方法进行链路预测,并取得了具有竞争力的结果<sup>[12]</sup>。

此类方法有效避免了相似度假设,使得性能有所提高,但其不足之处在于:1)此类人为定义的特征受主观意识的影响,导致选取的部分特征不一定有利于模型性能的提高,例如,对于某些热点新闻、消息,其转发过程很可能与年龄、性别等特征无关;2)此类模型的特征提取过程需要耗费大量的人力与时间,而且模型的普适性普遍较低。例如在微博消息传播网络中定义的特征向量,即便是在同类社交网络(如邮件网络、论文合作关系等)中依然很难迁移。

近年来,通过图嵌入学习进行链路预测的算法层出不穷,其核心思想是从高维网络中抽取局部和全局结构信息,学习一个映射函数,将节点表示为低维空间的一个向量<sup>[13]</sup>。图嵌入学习能够很好地解决图数据难以高效输入机器学习算法中的问题。

基于随机游走的图嵌入算法如DeepWalk<sup>[14]</sup>,Node2vec<sup>[15]</sup>等,将节点看作上下文,利用自然语言处理中的Word2vec<sup>[16]</sup>算法生成节点的向量表示。此类算法有效地改善了人为定义特征的局限性,提高了链路预测的精确度,同时也具有较高的普适性。但是,此类算法大多没有监督信息,很难根据实际的预测效果对生成的节点向量进行修正。另外,采用随机游走的方法会导致算法的精确度不稳定。

基于矩阵分解的图嵌入算法,如HOPE<sup>[17]</sup>,ProNe<sup>[18]</sup>等,通常采用奇异值分解方法来获得节点的向量表示。此类算法有效地避免了基于随机游走的图嵌入算法因随机性带来的不利影响,但是在进行奇异值分解的过程中,其将依赖网络的

邻接矩阵或节点相似度矩阵,导致此类算法的时间复杂度与空间复杂度较高,难以在大型网络中使用。

近年来,基于图神经网络的图嵌入算法如 GCN(Graph Convolutional Network)<sup>[19]</sup>、GraphSage<sup>[20]</sup>等,利用深度学习算法高效的数据拟合能力来生成节点的向量表示。此类算法可以通过无监督、半监督和有监督的方式进行训练,极大地提高了链路预测的精确度与灵活性。但是,此类算法的时间复杂度和空间复杂度通常较高,无法适用于大型网络。特别是在工业网络中,节点和边常常需要局部更新,使得模型需要重新训练。GraphSage 算法通过小批量采样的方式解决了 GCN 需要在整个图上进行训练的问题,既解决了 GCN 和 GAT(Graph Attention Network)复杂度过高的问题,又解决了工业应用中无法适应网络更新变化的问题。基于 GraphSage 的算法<sup>[21]</sup>实现了工业级大规模推荐系统应用,效果显著,但不可避免的是其精确度较 GCN 具有一定损失。GAT<sup>[22]</sup>通过引入节点间注意力机制,进一步提高了精确度,然而带来的问题是复杂度大幅增加。此外,GAT 算法还需要节点自身具有特征,而真实世界的绝大多数网络中的节点不具备高维节点特征。为了解决以上问题,本文提出了融合快速注意力机制的节点无特征网络链路预测算法。

### 3 融合快速注意力机制的节点无特征网络链路预测算法

#### 3.1 图注意力结构

令  $G=(V,E,H)$  代表一个具有节点特征属性的无向图,其中  $V$  代表节点集, $E$  代表边集, $H$  代表节点特征集。图注意力层的输入为原始节点特征  $H \in \mathbb{R}^{n \times f}$ ,输出为节点向量  $H' \in \mathbb{R}^{n \times f'}$ ,其中  $n$  代表节点个数, $f$  代表原始特征向量维数, $f'$  代表输出节点向量维数。

首先计算两个节点  $i, j \in V$  之间的注意力系数  $a_{ij}$ ,其计算方法如式(1)所示:

$$a_{ij} = \frac{\exp(\text{LeakyReLU}(\vec{a}^T [\mathbf{W}\vec{h}_i \parallel \mathbf{W}\vec{h}_j]))}{\sum_{k \in N_i} \exp(\text{LeakyReLU}(\vec{a}^T [\mathbf{W}\vec{h}_i \parallel \mathbf{W}\vec{h}_k]))} \quad (1)$$

其中, $\vec{a} \in \mathbb{R}^{2f}$ , $\mathbf{W} \in \mathbb{R}^{f \times f'}$  为待学习参数, $N_i$  代表节点  $i \in V$  的一阶邻居节点集。

#### 3.2 快速注意力结构

对于原始的图注意力结构,采用向量化深度学习框架计算  $H'$ ,则计算  $\vec{a}^T [\mathbf{W}\mathbf{H} \parallel \mathbf{W}\mathbf{H}]$  的空间复杂度为  $O(n^2 f')$ ,时间复杂度为  $O(nf f' + n^2 f')$ 。为解决该问题,本文采用基于注意力的内存寻址策略<sup>[23]</sup>,其计算式如式(2)所示:

$$\omega_i(i) \leftarrow \frac{\exp(\beta K[k_i, \mathbf{M}_i(i)])}{\sum_j \exp(\beta K[k_i, \mathbf{M}_i(j)])} \quad (2)$$

其中, $K$  代表相似度指标,文献<sup>[23]</sup>中采用了余弦相似度,其计算式如下:

$$K[\mathbf{u}, \mathbf{v}] = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \cdot \|\mathbf{v}\|} \quad (3)$$

该计算方法有效避免了矩阵拼接,减少了内存空间的占用,同时减少了非线性激活函数  $\text{LeakyReLU}$  的计算量。本文在图数据中将其进行推广,两个节点  $i, j \in V$  间的注意力系数

$a_{ij}$  的计算式如下:

$$a_{ij} = \frac{\exp(\vec{a}^T \text{Similarity}(\mathbf{W}\vec{h}_i, \mathbf{W}\vec{h}_j))}{\sum_{k \in N_i} \exp(\vec{a}^T \text{Similarity}(\mathbf{W}\vec{h}_i, \mathbf{W}\vec{h}_k))} \quad (4)$$

改进后的  $a_{ij}$  其空间复杂度有所降低,但是逐个节点计算相似度仍然具有较高的时间复杂度与空间复杂度。为了降低算法复杂度,本文采用只计算重要节点间的注意力系数的策略,忽略非重要节点的计算。记重要节点的个数为  $m$ ,则可以计算出相似度矩阵  $\text{Sim}_{n \times m}$ ,同时为了提高预测精度,增加待学习参数  $\vec{a}^T$  的参数量,将其替换为待学习参数  $L_{m \times n}$ ,则注意力矩阵  $a_{ij} \in A_{n \times n}$  可以通过式(5)计算得到。本文为了方便实验,首先取节点度排名前 20% 的节点为重要节点,其他节点为非重要节点。

$$\mathbf{A} = \text{Sim}L \quad (5)$$

其次对注意力矩阵  $a_{ij} \in A_{n \times n}$  进行行标准化:

$$a_{ij} = \frac{\exp(a_{ij})}{\sum_{1, n} \exp(a_m)} \quad (6)$$

则优化后的式(6)计算过程中需要占用内存最大规模的矩阵为  $A_{n \times n}$ ,故空间复杂度由  $O(n^2 f')$  降低至  $O(n^2)$ ,且不需要网络中的所有节点两两之间计算相似度系数,仅需要网络中所有节点  $n$  个与前  $m$  个重要节点计算,故时间复杂度由  $O(n^2)$  降低至  $O(mn)$ 。

然后计算单注意力层输出的节点向量  $\vec{h}_i' \in \mathbb{R}^{f'}$ ,其计算式如下:

$$\vec{h}_i' = \sigma(\sum_{j \in N_i} a_{ij} \mathbf{W}\vec{h}_j) \quad (7)$$

最后,多注意力层节点输出的节点向量  $\vec{h}_i' \in \mathbb{R}^{d'}$ ,其计算式如下:

$$\vec{h}_i' = \parallel_{k=1}^K \sigma(\sum_{j \in N_i} a_{ij}^k \mathbf{W}^k \vec{h}_j) \quad (8)$$

其中, $K$  代表注意力层堆叠层数, $\parallel$  代表行向量拼接运算, $N_i$  为节点  $i \in V$  的一阶邻居集合, $\vec{h}_i' \in \mathbf{H}'$  即为最终得到的节点  $i \in V$  的向量表示。

#### 3.3 算法扩展

对于节点无特征网络  $G=(V,E)$ ,其节点集并没有对应的特征集  $\mathbf{H}$ 。特征集  $\mathbf{H}$  的本质是通过向量的形式描述节点本身所具有的特征属性,例如,购物网络中的商品节点具有产地、销量、价格等特征属性。在节点无特征网络中,节点特征属性稀少且不规整。现有的扩展方法大多是指定网络的邻接矩阵为特征集  $\mathbf{H}$ ,然而邻接矩阵通常维数较大,不利于模型训练,而且矩阵元素稀疏,含有大量的数据噪声。本文认为节点的局部结构也是节点所具有的特征,解决该问题的关键在于节点局部结构的向量化描述。鉴于 Node2vec 算法随机游走过程可控、输出稳定、算法运行效率高等优点,因此,本文利用其去除节点局部结构的数据噪声,将节点的局部结构转化为对应的向量表示,最终得到算法的输入特征集  $\mathbf{H}$ 。

#### 3.4 链路预测

如图 1 所示,对于节点无特征网络  $G=(V,E)$ ,首先,将  $E$  中所有元素看作正例样本,记为  $S_1$ ,从  $V$  中随机抽取节点对  $v_i, v_j$  组成连边  $e_{ij}$ ,若  $e_{ij} \notin E$ ,则将其加入反例样本集  $S_0$ ,重复

该步骤直到  $|S_0| = |S_1|$ , 则得到样本空间  $S = S_0 \cup S_1$ 。

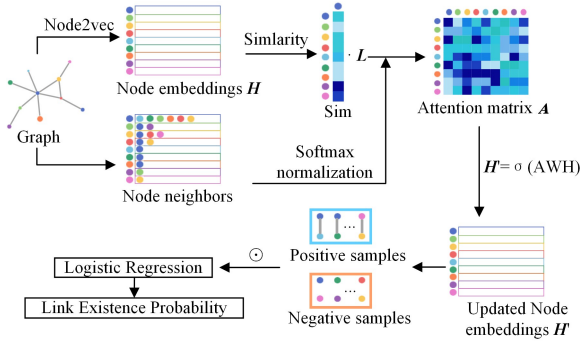


图1 FALP 算法的流程图

Fig. 1 Flow chart of FALP algorithm

其次,通过 Node2vec 算法在网络中进行有偏随机游走,得到节点局部结构对应的向量表示,将其作为算法的输入特征集  $H$ 。

本文为了方便实验,取节点度排名前 20% 的节点为重要节点,其他节点为非重要节点。接着,计算网络中所有节点与 Top- $m$  个重要节点的相似度,通过节点相似度矩阵  $\mathbf{Sim}$  与待学习参数  $L$  得到未归一化的注意力系数矩阵  $\mathbf{A}$ ,根据节点的邻居信息对得到的注意力系数进行 Softmax 归一化,从而得到注意力系数矩阵  $\mathbf{A}$ 。

最后通过式(7)或式(8)进行迭代训练得到节点的向量表示矩阵  $H'$ 。根据样本空间中对应的节点对计算哈达玛积,得到其对应的连边向量,通过逻辑回归分类器得到连边存在的概率。FALP 算法的主要实现过程如算法 1 所示。

#### 算法 1 FALP

输入:图  $G=(V,E)$ ,注意力邻居聚合层堆叠层数  $K$ ,重要节点数目  $m$ ,节点表示向量输出维度  $\text{dim}$ ,随机游走步长  $\text{Len}$ ,深度优先随机游走控制参数  $d$ ,广度优先随机游走控制参数  $b$ ,迭代次数  $\text{epochs}$   
输出:最优模型  $M$

1. 将图  $G$  中存在的所有连边作为正例样本
2. 随机不放回选取等量的在网络中不存在的连边作为反例样本
3. 将正例样本与反例样本合并打乱,根据比例确定训练集和测试集
4. 从图  $G$  中选择  $m$  个重要节点
5.  $\mathbf{H} = \text{Node2vec}(G, \text{dim}, \text{Len}, d, b)$
6. for  $e=1, 2, \dots, \text{epochs}$  do
7. for  $k=1, 2, \dots, K$  do
8.  $\mathbf{Sim}^k = \text{Similarity}(\mathbf{W}^k \mathbf{H}^k, (\mathbf{W}^k \mathbf{H}^k)_m)$
9.  $\mathbf{Attention}^k = \text{softmax}(\mathbf{Sim}^k L)$
10.  $\mathbf{H}^{k'} = \text{elu}(\mathbf{Attention}^k \mathbf{W}^k \mathbf{H}^k)$
11. end for
12.  $\mathbf{H} = \prod_{k=1}^K \mathbf{H}^{k'}$
13. 根据得到的节点向量表示矩阵  $\mathbf{H}$ ,通过哈达玛积计算训练集中连边对应的向量表示
14. 将连边及对应的向量表示通过逻辑回归分类器分类,得到连边存在的概率
16. 通过二元交叉熵损失函数计算损失
17. 在测试集上计算 AUC 和 Precision 指标,保存模型
18. end for
19. 选择最优模型  $M$  并返回

## 4 实验例证

### 4.1 数据

数据集的详细信息如表 1 所列。

表 1 数据集信息  
Table 1 Datasets information

| Name    | Public Dataset |          |        | Private Dataset |
|---------|----------------|----------|--------|-----------------|
|         | BlogCatalog    | WikiVote | Email  | CAFN            |
| # nodes | 10 312         | 7 115    | 1 005  | 20 746          |
| # edges | 333 983        | 103 689  | 25 571 | 136 448         |

(1)BlogCatalog<sup>[24]</sup>是一个博客社交网络,其中节点代表一个用户,边代表用户间的好友关系。

(2)WikiVote<sup>[25]</sup>是一个投票选举网络,其中节点代表维基百科的用户,边代表用户之间的投票关系。

(3)Email<sup>[26]</sup>是一个欧洲邮件网络,其中节点代表用户,边代表用户之间发送邮件的关系。

(4)CAFN(Collective Attention Flow Network)<sup>[27]</sup>是一个 WEB 站点点击流网络,其中节点代表 WEB 站点,边代表用户在 WEB 站点间的跳转。

### 4.2 评价指标

AUC 值的计算式如式(9)所示。首先,初始化  $n_0 = 0$ ,  $n_1 = 0$ ;其次,将测试集划分为  $E_0$  和  $E_1$  两个集合,其中  $E_0$  中的边在网络中不存在,  $E_1$  中的边在网络中存在;然后,进行一次随机实验,分别从  $E_0$  和  $E_1$  中有放回地抽取节点对  $e_0$  和  $e_1$ ,通过模型得到  $e_0$  和  $e_1$  的概率,分别记为  $p_0$  和  $p_1$ ,若  $p_1 > p_0$ ,则  $n_1$  自增 1,若  $p_1 = p_0$ ,则  $n_0$  自增 1;最后,重复该随机实验  $N$  次,可以得到链路预测算法的 AUC 值为:

$$AUC = \frac{n_1 + 0.5 \times n_0}{N} \quad (9)$$

Precision 值的计算式如式(10)所示。通过模型得到测试集中节点对在网络中存在的概率  $P$ ,按照概率值从大到小排序,如果前  $L$  个节点对中有  $l_1$  个节点对之间的连边预测正确,则可以得到链路预测算法的 Precision 值为:

$$Precision = \frac{l_1}{L} \quad (10)$$

### 4.3 基准算法

本文选取了近年来提出的 5 种基准算法与 FALP 算法进行对比实验。

(1)DeepWalk:DeepWalk 通过在图中随机游走得到节点序列,接着利用 Word2vec 算法得到节点的向量表示。

(2)Hope:Hope 通过广义奇异值分解分解相似度矩阵得到节点的向量表示。

(3)Line<sup>[28]</sup>:Line 定义一阶相似度与二阶相似度,然后根据特定的优化函数得到节点的向量表示。

(4)Node2vec:Node2vec 是对 DeepWalk 的改进算法,该算法通过在图中进行有偏随机游走得到节点序列,接着通过 Word2vec 算法得到节点的向量表示。

(5)ProNe:ProNe 是一种基于矩阵分解和频谱传播的图嵌入算法,该算法通过稀疏矩阵分解得到节点的初始向量表示,接着利用高阶切比雪夫不等式对网络的频谱空间进行调整,得到节点的最终向量表示。

### 4.4 实验参数设置

为了便于对比分析,实验中所有算法生成的节点向量

维数均设置为128,其他参数值遵循原文中的设定值。具体参数设置如下:在FALP算法中,Node2vec模块随机游走步长为10,随机游走步数为80,参数 $b$ 为0.25,参数 $d$ 为2,窗口大小为5,Attention模块隐藏层节点个数为128,注意力层数为2,dropout参数为0.6,相似度计算策略选择余弦相似度,重要节点为网络中度排名top 20%的节点,Epochs为100,学习率为 $5 \times 10^{-4}$ 。Node2vec算法和DeepWalk算法的所有参数与FALP保持一致。Line算法采用2阶相似度进行训练。ProNe算法中切比雪夫不等式展开项数为5,位置参数为0.2,尺度参数为0.5。所有算法均使用数据集中90%的数据进行训练,使用10%的数据进行测试。

#### 4.5 实验分析

不同算法的AUC值随迭代次数增加的变化过程如图2所示,可以看出,相比对比算法,FALP算法在BlogCatalog, WikiVote和CAFN数据集上的预测性能显著提高,而且其收敛速度明显快于对比算法;在Email数据集上其预测性能略低于Hope算法,原因在于Email网络结构相对稀疏,导致样本数量过少,不利于数据拟合。基于矩阵分解的图嵌入算法ProNe在该数据集上略有优势。

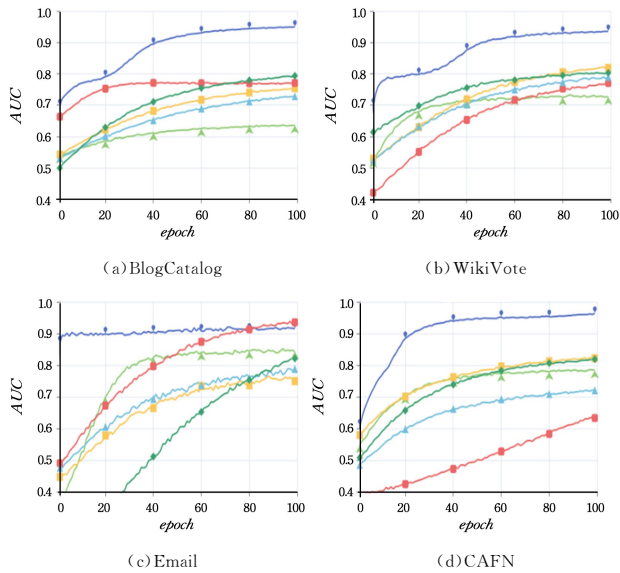


图2 训练过程中AUC值变化曲线

Fig. 2 AUC varying curve of different algorithms in different datasets

为了使实验数据更具说服力,本文重复了10次实验并列出了其AUC的平均值及相对误差(见表2)。从表2可以看出,FALP算法不仅在预测指标中优势明显,而且其稳定性普遍高于对比算法。ProNe和Hope矩阵分解类算法虽然在Email数据集上略有优势,但在较大的数据集上,其预测性能较差,而且其稳定性在对比算法中是最差的。

在AUC值相差不大的情况下,常利用Precision指标来评价链路预测算法,其值越高,算法综合性能越好。分别取参数 $L$ 的值为测试集大小的10%,20%,...,90%,100%,其对应的Precision值变化曲线如图3所示。结果表明,FALP算法的Precision值曲线下降速度最缓,稳定性较高,且在不同数据集中均优于对比算法。

表2 10次实验的平均AUC和相对误差

Table 2 Average AUC and relative error in 10 experiments (单位:%)

|             | FALP       | DeepWalk   | Hope       | Line       | Node2vec   | ProNe      |
|-------------|------------|------------|------------|------------|------------|------------|
| BlogCatalog | 95.08±0.06 | 74.96±0.95 | 72.76±3.00 | 63.29±0.21 | 75.21±1.09 | 76.71±8.42 |
|             | 93.63±0.06 | 81.03±1.06 | 78.31±2.01 | 73.18±0.14 | 79.25±0.77 | 79.36±1.12 |
| WikiVote    | 92.07±0.06 | 75.87±1.06 | 91.87±4.03 | 85.19±0.26 | 79.19±0.98 | 94.41±2.45 |
|             | 0.19       | 1.07       | 1.07       | 0.26       | 0.98       | 2.45       |
| Email       | 96.05±0.24 | 82.33±0.78 | 67.21±1.00 | 78.78±0.30 | 72.59±0.53 | 83.21±2.17 |
|             | 0.24       | 0.78       | 1.00       | 0.30       | 0.53       | 2.17       |

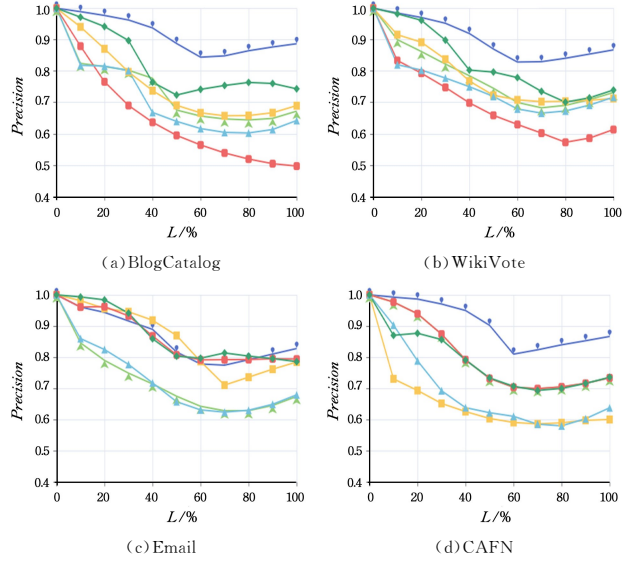


图3 不同L值对应的Precision值变化曲线

Fig. 3 Precision varying curve corresponding to different L

**结束语** 本文受内存寻址策略的启发,结合复杂网络中重要节点发现的相关工作,设计了一种快速高效的图注意力卷积结构,提出了节点无特征网络的链路预测算法FALP。

FALP算法的主要优点如下:1)相比传统的注意力卷积结构,其在注意力系数计算过程中不仅避免了三维矩阵运算,大幅度减少了内存占用,而且避免了网络中所有节点两两之间计算注意力系数,时间复杂度由平方阶降至线性阶;2)通过随机游走的预处理策略,将其在节点无特征网络中进行了推广,扩大了图神经网络类算法的使用范围;3)扩展性较强,稍加改动即可用于节点分类、社区检测、重要节点发现等复杂网络中的其他下游任务。

FALP算法的不足之处在于其对小型网络的网络结构变化不敏感,本文猜想其原因在于网络的幂律分布特性与本文选取的相似度和重要节点发现策略存在一定冲突。因此,在未来的研究工作中,将参考幂律分布的相关工作,优化FALP算法在小型网络中的预测效率。

#### 参考文献

- [1] LI M, MENG X M, ZHENG F X, et al. Identification of Protein Complexes by Using a Spatial and Temporal Active Protein Interaction Network [J]. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2020, 17(3): 817-827.
- [2] YASUNAGA M, KASAI J, ZHANG R, et al. A Large Annotated Corpus and Content-Impact Models for Scientific Paper Summarization with Citation Networks[C]// The AAAI Confe-

- rence on Artificial Intelligence. 2019;7386-7393.
- [3] MANGIONI G, JURMAN G, DEDOMENICO M. Multilayer Flows in Molecular Networks Identify Biological Modules in the Human Proteome[J]. *IEEE Transactions on Network Science and Engineering*, 2018, 7(1):411-420.
- [4] ZHOU F, YANG Q, ZHONG T, et al. Variational Graph Neural Networks for Road Traffic Prediction in Intelligent Transportation Systems[J]. *IEEE Transactions on Industrial Informatics*, 2021, 17(4):2802-2812.
- [5] LUO J W, LONG Y H. NTSHMDA: Prediction of Human Microbe-Disease Association Based on Random Walk by Integrating Network Topological Similarity[J]. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2020, 17(4):1341-1351.
- [6] CALDERONI F, CATANESE S, MEOP D, et al. Robust link prediction in criminal networks: A case study of the Sicilian Mafia[J]. *Expert Systems with Applications*, 2020, 161(3):113666.
- [7] HWANG D, YANG S, KWON Y, et al. Comprehensive Study on Molecular Supervised Learning with Graph Neural Networks[J]. *Journal of Chemical Information and Modeling*, 2020, 60(12):5936-5945.
- [8] MOLZAHND K, WANG J. Detection and Characterization of Intrusions to Network Parameter Data in Electric Power Systems[J]. *IEEE Transactions on Smart Grid*, 2019, 10(4):3919-3928.
- [9] WU S W, ZHANG W T, SUN F, et al. Graph Neural Networks in Recommender Systems: A Survey [EB/OL]. (2020-11-04) [2021-08-30]. <https://arxiv.org/abs/2011.02260>.
- [10] KUMAR A, SING H S, SINGH K, et al. Link prediction techniques, applications, and performance: A survey[J]. *Physica A: Statistical Mechanics and its Applications*, 2020, 553(6):124289.
- [11] ZHANG M H, CHEN Y X. Link Prediction Based on Graph Neural Networks[C]// *The Advances in Neural Information Processing Systems 31 (NeurIPS)*. 2018:5165-5175.
- [12] ZHANG M, LIANG Y Y, HUANG X J. Link prediction and analysis of formation mechanism of complex networks based on ensemble learning[J]. *Journal of Chongqing University of Posts and Telecommunications(Natural Science Edition)*, 2020, 32(5):759-768.
- [13] GOYAL P, FERRARAE. Graph Embedding Techniques, Applications, and Performance: A Survey[J]. *Knowledge-Based Systems*, 2017, 151:78-94.
- [14] PEROZZI B, AL-RFOU R, SKIENA S. Deepwalk: Online learning of social representations[C]// *20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2014:701-710.
- [15] GROVER A, LESKOVEC J. node2vec: Scalable Feature Learning for Networks[C]// *22th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016:855-864.
- [16] MIKOLOV T, CHEN K, CORRADO G S, et al. Efficient Estimation of Word Representations in Vector Space[C]// *International Conference on Learning Representations (ICLR)*. 2013.
- [17] OUM D, CUI P, PEI J, et al. Asymmetric Transitivity Preserving Graph Embedding[C]// *22th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016:1105-1114.
- [18] ZHANG J, DONG Y, WANG Y, et al. ProNE: Fast and Scalable Network Representation Learning[C]// *28th International Joint Conference on Artificial Intelligence (IJCAI)*. 2019:4278-4284.
- [19] KIPF T, WELING M. Semi-Supervised Classification with Graph Convolutional Networks[C]// *International Conference on Learning Representations (ICLR)*. 2016.
- [20] HAMILTON W, YING R, LESKOVEC J. Inductive Representation Learning on Large Graphs[C]// *31th Conference on Neural Information Processing Systems (NeurIPS)*. 2017:1024-1034.
- [21] YING R, HE R, CHEN K, et al. Graph Convolutional Neural Networks for Web-Scale Recommender Systems[C]// *24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2018:947-983.
- [22] VELICKOVIC P, CUCURULL G, CASANOV A A, et al. Graph Attention Networks[C]// *International Conference on Learning Representations (ICLR)*. 2018.
- [23] GRAVES A, WAYNE G, DANIELKA L. Neural Turing Machines [EB/OL]. (2014-12-10) [2021-08-30]. <https://arxiv.org/abs/1410.5401>.
- [24] TANG L, LIU H. Relational learning via latent social dimensions[C]// *15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2009:817-826.
- [25] LESKOVEC J, HUTTENLOCHER D, KLEINBERG J. Predicting Positive and Negative Links in Online Social Networks[C]// *19th International Conference on World Wide Web (WWW)*. 2010:641-650.
- [26] YIN H, BENSON A, LESKOVEC J, et al. Local Higher-Order Graph Clustering[C]// *23th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2017:555-564.
- [27] LI Y, MENG X F, ZHANG Q, et al. Common patterns of online collective attention flow[J]. *Science China Information Sciences*, 2017, 60(5):059102.
- [28] TANG J, QU M, WANG M Z, et al. LINE: Large-scale Information Network Embedding[C]// *24th International Conference on World Wide Web (WWW)*. 2015:1067-1077.



**LI Yong**, born in 1979, Ph.D, associate professor, is a member of China Computer Federation. His main research interests include social computing and data analytics.



**WU Jing-peng**, born in 1998, postgraduate. His main research interests include machine learning, link prediction in complex networks and graph neural networks.