文章编号:1671-6833(2024)06-0075-08

一种近似图神经网络框架的无监督链路预测算法

李格格[1,2], 冶忠林[1,2], 曹淑娟[1,2], 周 琳[1,2], 王雪力[1,2]

(1.青海师范大学 计算机学院,青海 西宁 810008;2.青海师范大学 藏语智能信息处理及应用国家重点实验室,青海,西宁 810008)

摘 要:对于无标签网络,由于基于图神经网络的链路预测方法使用其高效建模机制进行链路预测任务时性能较差,因此,提出了一种近似图神经网络框架的无监督链路预测算法(ALIP),旨在模拟图神经网络算法的高效建模机制和学习过程,解决网络节点标签缺失导致的建模不充分问题。首先,参照 GCN 的输入层,融合网络的结构信息和节点属性;其次,使用矩阵分解替代 GCN 的隐藏层,模拟正向传播;再次,借鉴恒等映射和高阶近邻的思想实现向量转化和模型优化,从而得出网络节点表示向量,该过程模拟 GCN 的反向传播;最后,计算相似度矩阵,进行链路预测任务性能评测。在 Citeseer 数据集、DBLP 数据集和 Cora 数据集上的实验结果表明:所提 ALIP 算法 *AUC* 值最高为 98.01%,其性能优于其他 23 种链路预测算法,证明了该算法的有效性和可行性,同时也为无标签的复杂网络链路预测任务提供了一种新的解决方案。

关键词:矩阵分解; 向量优化; 图卷积神经网络; 相似度矩阵; 链路预测; 高阶近邻

中图分类号: TP393.0 **文献标志码**:A **doi**:10.13705/j.issn.1671-6833.2024.03.011

现实世界中,事物之间的关系极其复杂,而复杂网络可以恰当描述此类关系。根据实体类型,复杂网络可被分为社交分析网络[1-3]、蛋白质分子网络[4]、通信网络[5]等。分析预测网络中未连边节点之间的关系可促进各个相关领域的快速发展,上述预测节点关系的过程即为链路预测[6](link prediction, LP)。

现如今,链路预测的研究方法有基于启发信息的链路预测方法[7]、基于机器学习的链路预测方法、基于网络嵌入的链路预测方法等。

基于启发信息的链路预测方法中,通常会使用 CN(common neighbors)指标[8]、RA(resource allocation)指标[8]、AA(adamic-adar)指标[8]、Jaccard 指标[9]、Sorenson 指标[10]、Salton 指标[8]、LP(local paths)指标[11]、Katz 指标[12]、LHN-Ⅱ指标[13]、cos+指标[14]、SRW 指标[11]等。基于启发信息的链路预测方法对于节点间是否有边存在着很大的假设性,倘若该假设错误,便会导致失败[15]。同时,该类方法对网络结构信息的表示较差,并且多数链接假设均只适用于社交网络,在非社交网络中预测性能不佳。

基于机器学习的链路预测方法中,通常使用的分类方法有支持向量机(support vector machine, SVM)、决策树(decision tree, DT)等。廖亮等[16]提出了一种基于 SVM 的机会网络链路预测,考虑了空间相似性和时间特征。杨妮亚等[17]提出了一种基于聚类和决策树的链路预测(CDTLinks),以网络中 2 种类型对象相互作为对象的方法去获得对象的特征表示,并且分别进行聚类,同时对双类型异质网络构建决策树,由信息增益作为选取不同分支的标准,结合上述 2 种方法判断不同类型节点间的连接关系,最后验证 CDTLinks 方法的有效性。基于机器学习的链路预测方法尽管有不错的效果,但是会占用大量的存储空间,且计算时间长。

基于网络嵌入的链路预测方法中会使用的技术有随机游走、矩阵分解、图卷积神经网络[18](graph convolutional neural network, GCN)等。Lei 等[19]通过 GCN、LSTM、GAN 提取加权网络中的非线性特

收稿日期:2023-10-11;**修订日期**:2023-11-20

基金项目:国家重点研发计划(2020YFC1523300);青海省创新平台建设项目(2022-ZJ-T02)

通信作者:冶忠林(1989—),男,青海民和人,青海师范大学教授,博士,博士生导师,主要从事图神经网络及网络表示学习研究,E-mail: zhonglin_ye@foxmail.com。

引用本文:李格格,冶忠林,曹淑娟,等. 一种近似图神经网络框架的无监督链路预测算法[J]. 郑州大学学报(工学版),2024,45(6):75-82. (LI G G, YE Z L, CAO S J, et al. An unsupervised link prediction algorithm based on an approximate graph neural network framework[J]. Journal of Zhengzhou University (Engineering Science),2024,45(6):75-82.)

征,同时通过 GCN 提取各个快照的局部特征,通过 LSTM 提取网络的演化特征,进而使得该算法链路预测性能优异。首先,GCN 将网络结构的边信息和节点信息输入到输入层中;其次,通过输出层输出的预测标签结果与网络的节点标签计算损失;最后,进行反向传播和梯度更新。在不断的迭代训练过程中得到最优模型。对于有监督任务,基于 GCN 的链路预测方法虽然表现出优异的性能,但是会存在耗费时间、资源等问题。而现实生活中存在大量的无标签网络,当使用基于 GCN 的链路预测方法进行处理时效果较差,针对上述问题,本文提出了一种近似图神经网络框架的无监督链路预测算法(an unsupervised link prediction algorithm based on an approximate graph neural network framework, ALIP)。

本文创新点有以下 3 点。

(1)ALIP 算法模拟了 GCN 训练的过程,对比 GCN 的输入层,该算法的输入层有效保证了网络结构特征的完整性。

(2)使用矩阵分解替代 GCN 的隐藏层,模拟 GCN 中的正向传播。

(3)借鉴深度图神经网络(DeeperGCN)[20-21]中的恒等映射思想和 NEU 算法[22]中的高阶近邻的思想,实现算法优化问题,避免网络特征建模不充分的问题。

1 GCN 算法框架

图结构不具备平移不变性,图中节点的周围结构通常不相同,对于该类数据的处理,传统神经网络框架如卷积神经网络(convolutional neural network, CNN)、循环神经网络(recurrent neural network, RNN)则会失效,无法获得较优的效果。目前常用的方法有 DeepWalk、node2vec、GNN、GCN 等,其中 GCN 与 CNN 均为特征提取器,只是所研究的对象不同。通过 GCN 对图结构数据进行特征提取所得到的特征可用于众多下游任务,如边预测、节点分类等。GCN 总体框架图如图 1 所示。

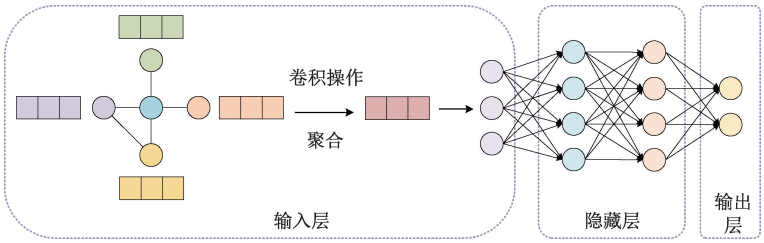

图 1 GCN 框架图

Figure 1 GCN framework diagram

由图 1 可知,GCN 由三部分组成,分别为输入层、隐藏层、输出层。在输入层中,图结构数据中每个节点的节点特征均需融合自身特征信息以及其邻接节点的特征信息,其中节点自身特征为矩阵 $\boldsymbol{X}$,包含网络结构信息的邻接矩阵为 $\boldsymbol{A}$。在图 1 中,GCN 用平均池化的方法模拟卷积操作,即将节点特征向量进行平均操作,然后将最终计算得到的平均值输入到隐藏层中进行计算。在隐藏层中,层与层之间的计算过程为

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

式中:$\sigma(\cdot)$为非线性激活函数;$\tilde{\boldsymbol{A}}$ 为含有自连接的邻接矩阵,是网络节点之间的关系特征矩阵 $\boldsymbol{A}$ 与单位矩阵 $\boldsymbol{I}$ 相加的结果;$\tilde{\boldsymbol{D}}$ 为 $\tilde{\boldsymbol{A}}$ 对应的度矩阵;$\boldsymbol{H}$ 为每一层的节点特征。

2 ALIP 算法框架

图结构通常是不规则的,图神经网络的处理对象为图结构数据,GCN 是图神经网络的代表算法。现有的研究结果表明,图神经网络使用半监督机制,在链路预测、节点分类和推荐系统等机器学习任务中表现非常好。对于无标签网络,使用 GCN 的建模机制进行链路预测时效果较差,但是现实生活中,大多图结构数据无标签信息,因此,本文提出了 ALIP 算法。

ALIP 算法模拟两层的 GCN 框架,使用矩阵分解替代 GCN 中的正向传播,使用高阶近邻和恒等映射的思想替代 GCN 中的反向传播,起到更新参数的作用。ALIP 算法框架如图 2 所示。

由图 2 可知,ALIP 算法分输入层、计算层、输出层。本文使用 *AUC* 指标评测 ALIP 算法性能,以此证明该算法的有效性和可行性。

(1)输入层:输入网络的结构信息和节点属性。ALIP 算法输入层参照了 GCN 的输入层,将关系特征矩阵 $\boldsymbol{A}$ 和节点属性 $\boldsymbol{X}$ 输入到算法中,即将式(1)中的 $\boldsymbol{HW}$ 神经元计算过程替换为 $\boldsymbol{X}$,则提出的输入层的特征为

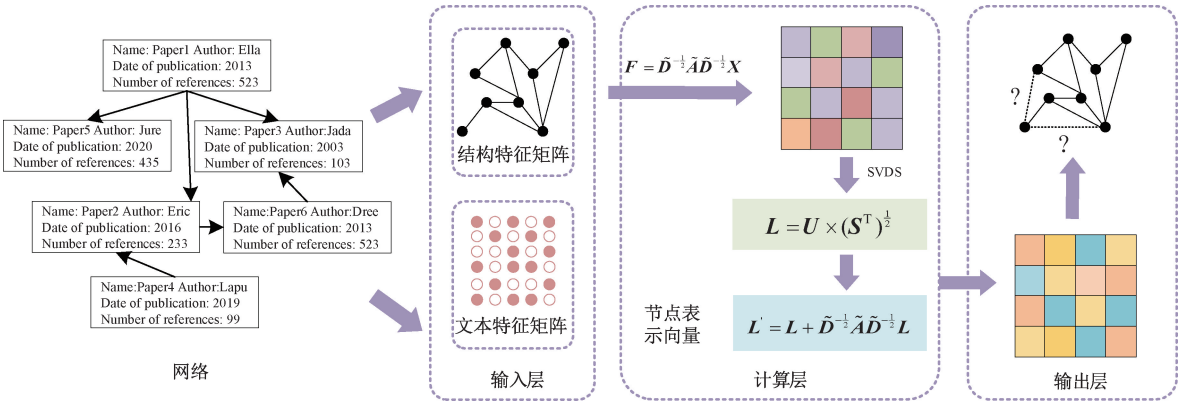

图 2　ALIP 算法框架

Figure 2　ALIP algorithm framework

$$\boldsymbol{F}=f(\boldsymbol{X},\boldsymbol{A})=\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{X}。\tag{2}$$

式中:$\boldsymbol{A}$ 为网络结构的邻接矩阵;$\tilde{\boldsymbol{A}}$ 为含有自连接的邻接矩阵,保留了网络结构的自身特征;$\tilde{\boldsymbol{D}}$ 为含有自连接的度矩阵,是 $\tilde{\boldsymbol{A}}$ 对应的度矩阵。由于 ALIP 算法不进行参数更新及多次训练拟合,故不用学习神经元之间的权重矩阵 $\boldsymbol{W}$,也不用进行激活操作。综上,可知 ALIP 算法的输入本质上是借鉴了 GCN 结合图特征及节点自身特征融合的思想,来融合结构信息和节点属性信息。上述 $\boldsymbol{A}$ 与 $\tilde{\boldsymbol{A}}$ 的关系为

$$\tilde{\boldsymbol{A}}=\boldsymbol{A}+\boldsymbol{I}。\tag{3}$$

式中:$\boldsymbol{I}$ 为单位矩阵。

(2)计算层:对输入层所输出的向量矩阵进行矩阵分解和向量优化。浅层神经网络不断被使用,已经不能够较好地解决现有问题,深层神经网络方法使得问题能够得到较优解决,那么在实际生活中,深层神经网络的训练会遇到梯度弥散、梯度爆炸、网络退化等问题。梯度弥散和梯度爆炸的问题已经通过标准初始化和中间层正则化等优化方法得以有效控制。网络退化问题反映了深层网络不易优化的问题,因此,现有的工作主要是引入深层残差网络,融入恒等映射,从而将模型的深度加深,防止网络退化。残差单元连接方式如图 3 所示。

图 3 中跳层连接的残差单元为

$$\boldsymbol{F}'(\boldsymbol{x})=\boldsymbol{F}(\boldsymbol{x})+\boldsymbol{x}。\tag{4}$$

式中:$\boldsymbol{x}$ 为输入的原始特征;$\boldsymbol{F}(\boldsymbol{x})$ 为残差函数,定义了原始神经结构的输入 $\boldsymbol{x}$,也是一个递归的过程;$\boldsymbol{F}'(\boldsymbol{x})$ 为跳层连接的残差单元的输出。

由图 3 可知,残差神经网络主要是在不同层之间插入原始的特征输入,即进行了恒等映射,从而避免随着模型越来越深而出现网络退化的问题。

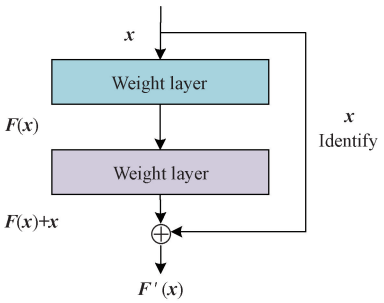

图 3　跳层连接的残差单元

Figure 3　Residual units connected by jumpers

在网络表示学习领域,为了解决低阶网络不能建模网络节点之间的高阶关系的问题,研究人员提出了高阶近邻的方法,一类方法是在建模过程中加入节点之间的高阶关系;另外一类方式是将低阶模型学习得到的节点表示向量转化为高阶表示向量。其过程为

$$\begin{cases}\boldsymbol{R}'=\boldsymbol{R}+\lambda\boldsymbol{B}\cdot\boldsymbol{R};\\ \boldsymbol{C}'=\boldsymbol{C}+\lambda\boldsymbol{B}^{\mathrm{T}}\cdot\boldsymbol{C}。\end{cases}\tag{5}$$

式中:$\lambda\in\left(0,\frac{1}{2}\right]$;$\boldsymbol{B}$ 为归一化邻接矩阵;$\boldsymbol{R}$ 为低阶的网络表示向量;$\boldsymbol{C}$ 为上下文节点的表示向量。通过式(5)可以将低阶向量转化为高阶向量,从而实现表示向量的优化过程。

本文将残差网络中的恒等映射思想和高阶网络表示学习中的高阶近邻思想引入 ALIP 算法中。

首先,将输入层输出的结构特征矩阵 $\boldsymbol{F}$ 使用矩阵分解方法进行分解,该方法替代 GCN 中隐藏层的正向传播过程,即将其分解为 3 个矩阵相乘的形式:

$$\boldsymbol{F}=\boldsymbol{U}\times\boldsymbol{S}\times\boldsymbol{V}。\tag{6}$$

式中:矩阵 $\boldsymbol{U}$ 代表结构特征矩阵 $\boldsymbol{F}$ 对应的奇异向量;矩阵 $\boldsymbol{S}$ 代表结构特征矩阵 $\boldsymbol{F}$ 对应的对角矩阵,矩阵 $\boldsymbol{S}$ 中的对角元素为结构特征矩阵 $\boldsymbol{F}$ 的奇异值。

其次,将矩阵 $\boldsymbol{U}$ 和矩阵 $\boldsymbol{S}$ 进行结合计算,得到网络中节点的表示向量 $\boldsymbol{L}$:

$$\boldsymbol{L} = \boldsymbol{U} \times (\boldsymbol{S}^{\mathrm{T}})^{\frac{1}{2}}。 \tag{7}$$

最后,为了有效防止网络原有特征丢失,将节点表示向量 $\boldsymbol{L}$ 和原始的网络特征矩阵进行结合,结合式(4)、(5)提出式(8)。式(8)融合了残差网络中的恒等映射和高阶网络表示学习中的高阶向量转化的思想,使得最终节点表示向量得到了优化。该步骤替代 GCN 中反向传播的参数更新过程,具体形式为

$$\boldsymbol{L}' = \boldsymbol{L} + \tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{L}。 \tag{8}$$

式中:$\boldsymbol{L}'$ 代表计算层的输出。上述过程是一种无监督优化过程,其有别于 GCN 中的反向传播过程。本文 ALIP 算法向量优化计算量的增加来自于式(8),向量 $\boldsymbol{L}$ 的计算复杂度为 $O(|v|)$,其中 v 为网络节点个数,由于 $\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}$ 的计算复杂度为 $O(|v|^3)$,故 $\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{L}$ 的计算复杂度为 $O(|v|^4)$,综上 $\boldsymbol{L}'$ 的计算复杂度为 $O(|v|^4)$。

(3)输出层:根据计算层输出的矩阵,计算节点之间的相似度。根据计算层所输出的向量 $\boldsymbol{L}'$,计算节点对之间对应的相似度矩阵 $\boldsymbol{W}_{i,j}$,其具体过程为

$$\boldsymbol{W}_{i,j} = \boldsymbol{E}_i \times \boldsymbol{E}_j。 \tag{9}$$

式中:$\boldsymbol{E}_i$ 代表向量 $\boldsymbol{L}'$ 中的第 i 行向量;$\boldsymbol{E}_j$ 代表向量 $\boldsymbol{L}'$ 中的第 j 行向量。

在评测链路预测 ALIP 算法的过程中,基于计算所得到的相似度矩阵 $\boldsymbol{W}_{i,j}$,使用链路预测度量指标 *AUC* 评测本文所提链路预测算法的性能。本文算法步骤如下。

算法 1 ALIP 算法。

输入:网络 G 的边集合;训练比率 T;向量长度 K;

输出:*AUC* 值。

① 获取网络 G 的边集合;

② 计算出网络 G 节点数量 $|v|$;

③ 将网络 G 分为训练集和测试集,[训练集,测试集]←网络 G;

④ 初始化邻接矩阵 $\boldsymbol{A}$ 和文本矩阵 $\boldsymbol{X}$;

⑤ 初始化特征矩阵 $\boldsymbol{F}=\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{X}$;

⑥ 分解矩阵 $\boldsymbol{F}$ 为 $[\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}]=\mathrm{SVDS}(\boldsymbol{F}, \boldsymbol{k})$;

⑦ 获取表示向量 $\boldsymbol{L} = \boldsymbol{U} \times (\boldsymbol{S}^{\mathrm{T}})^{\frac{1}{2}}$;

⑧ 优化矩阵 $\boldsymbol{L}'=\boldsymbol{L}+\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\tilde{\boldsymbol{D}}^{-\frac{1}{2}}\boldsymbol{L}$;

⑨ 计算相似度矩阵 $\boldsymbol{W}_{i,j} = \boldsymbol{E}_i \times \boldsymbol{E}_j$;

⑩ 使用测试集计算 ALIP 算法的 *AUC* 值,*AUC*←训练集。

3 实验结果与分析

3.1 实验数据

本文使用 Citeseer 数据集、DBLP 数据集、Cora 数据集对 ALIP 算法进行验证,具体信息如表 1 所示。

表 1 数据集描述

Table 1 Datasets description

数据集	节点数	边数	平均度	密度	平均聚类系数
Citeseer	3 312	4 732	2.857	0.001	0.080
DBLP	3 119	39 516	21.070	0.005	0.221
Cora	2 708	5 429	4.010	0.001	0.130

Citeseer 数据集是引文网络,其中节点代表一篇论文,边代表论文相互之间的引用关系。DBLP 数据集是合作网络,其中节点代表作者,边代表作者之间的合作关系。Cora 数据集是引文网络,共存在 5 429 次引用。

由表 1 可知,3 个数据集节点数的差异并不明显,但是边的数量却有很大的差异,其中 DBLP 数据集的边数最多,会导致 DBLP 数据集的平均度、密度以及平均聚类系数均比 Citeseer 数据集和 Cora 数据集高出很多,进一步可推断 DBLP 网络属于稠密网络,Citeseer 网络和 Cora 网络属于稀疏网络。

3.2 评价指标

对于 ALIP 算法的实验性能的评估,选取 *AUC* 指标作为评判指标,该指标是根据链路预测算法输出的节点相似度矩阵进行计算的,是链路预测算法中的一个常用指标。整个过程是一个比较测试集中边相似值和不存在边相似值的过程,倘若边相似值大于不存在边的相似值,加 1 分,倘若二者相等,加 0.5 分。*AUC* 的值应为 0.5~1.0,其值越高,则该算法的精确度越高且性能越好。*AUC* 指标为

$$AUC = \frac{n_1 + 0.5n_2}{N}。 \tag{10}$$

式中:N 为独立比较次数;n_1 为边相似值大于不存在边相似值的次数;n_2 为边相似值等于不存在边的相似值的次数。

3.3 基准算法

本文选取了现有的 23 种链路预测算法作为基准算法与 ALIP 算法进行对比。基准算法涵盖了经典和最新的链路预测算法,如表 2 所示。

3.4 实验结果分析

为了确保算法对比的公平性,训练比率 T 设置为 0.7,0.8,0.9,网络节点表示向量长度为 100,并

表 2　基准算法总结表

Table 2　Benchmark algorithm summary table

算法类别	具体算法
基于局部信息的链路预测算法	CN[8]、Salton[8]、Jaccard[9]、HPI[8]、HDI[8]、LHN-Ⅰ[8]、AA[8]、RA[8]、PA[8]、LNBAA[23]、LNBCN[23]、LNBRA[23]
基于路径的链路预测算法	LP[11]、Katz[12]、LHN-Ⅱ[13]
基于随机游走的链路预测算法	ACT[24]、Cos+[14]、LRW[11]、SRW[11]
基于自洽相似性的链路预测算法	TSCN[25]
基于网络表示学习的链路预测算法	LPMF[26]、TELP[27]、NRLP[28]

且将 ALIP 算法运行 10 次的平均值作为该算法的实验结果,如表 3 所示。

由表 3 可知,在 Citeseer、DBLP、Cora 数据集上,仅有 Katz、LHN-Ⅱ、TELP、NRLP 算法的 *AUC* 均高于 90%,其中 Katz 算法和 LHN-Ⅱ算法属于基于路径的链路预测,TELP 算法和 NRLP 算法属于基于矩阵分解的链路预测,可见这两大类的链路预测方法性能较为优异,尤其是 Katz 算法和 NRLP 算法,在 Citeseer 数据集上的准确率达到了 97%。在 Citeseer 数据集上,当 $T=0.8$ 时,Katz 算法的 *AUC* 略高于 ALIP 算法,除此之外,ALIP 算法的 *AUC* 取得最高值,其最高值为 98.01%。而在 DBLP 数据集和 Cora 数据集上,ALIP 算法效果最好。

从算法复杂度的角度分析 ALIP 算法,只需将其与建模机制相同的 LPMF 进行对比,本次实验控制训练比率为 0.7,节点表示向量长度为 100,实验结果如表 4 所示。由表 4 可知,ALIP 算法时间比 LPMF 算法时间多 1~1.5 s,但 ALIP 算法相对于 LPMF 算法的性能有 1.08 百分点至 10.3 百分点的提升。综合考虑,在相同建模机制的算法下,ALIP 算法体现出一定的优越性。

综上所述,ALIP 算法可充分挖掘到网络的有效特征,同时能够保证网络特征相关信息的完整性,并且性能优异。

表 3　Citesser、DBLP 和 Cora 数据集上链路预测结果

Table 3　Link prediction results on Citesser, DBLP and Cora datasets

算法	*AUC*/%								
	Citeseer 数据集			DBLP 数据集			Cora 数据集		
	$T=0.7$	$T=0.8$	$T=0.9$	$T=0.7$	$T=0.8$	$T=0.9$	$T=0.7$	$T=0.8$	$T=0.9$
CN	68.13	72.08	74.67	85.49	88.40	90.68	69.50	72.38	78.19
Salton	66.32	72.73	74.44	86.00	87.92	90.74	69.38	72.13	77.89
Jaccard	66.51	72.25	74.33	85.92	88.26	90.98	69.25	72.00	77.09
HPI	66.29	72.18	74.42	85.61	88.95	90.77	69.38	72.44	77.93
HDI	66.03	72.52	74.17	85.72	88.31	90.84	69.52	72.53	76.67
LHN-Ⅰ	66.47	72.93	74.46	85.80	87.87	89.95	69.19	72.16	77.30
AA	66.37	72.22	74.33	86.00	88.22	90.95	69.35	72.66	77.60
RA	66.37	72.12	74.63	86.56	88.50	90.81	69.47	72.47	77.97
PA	78.98	79.06	79.53	76.39	77.13	77.54	71.50	71.91	71.50
LP	81.06	86.83	88.45	92.96	93.65	94.94	80.12	82.97	87.90
Katz	96.89	**97.98**	97.19	93.45	94.18	94.83	90.89	92.14	94.44
LHN-Ⅱ	95.76	96.85	96.20	90.86	91.80	92.80	89.41	90.37	93.64
LNBAA	66.37	72.64	74.52	86.07	88.42	91.12	69.42	72.50	78.01
LNBCN	66.70	72.27	74.25	85.60	88.47	90.80	69.50	72.19	77.79
LNBRA	66.05	72.23	74.27	85.86	88.91	91.23	69.32	72.84	77.74
ACT	75.88	75.59	73.79	79.00	80.07	80.84	74.11	73.67	74.00
Cos+	88.57	89.38	88.49	91.53	93.47	95.08	90.25	90.98	93.22
LRW	87.21	90.13	91.25	92.75	93.35	94.09	88.48	90.58	93.63
SRW	86.34	90.05	90.47	90.50	92.25	94.06	88.40	90.50	93.62
TSCN	84.26	85.68	86.27	91.25	91.03	92.34	88.35	90.64	92.98
NRLP	96.71	97.83	97.15	94.43	95.42	96.40	92.32	93.57	94.53
TELP	95.00	96.65	96.81	93.75	94.79	95.77	91.21	92.43	94.40
LPMF	87.18	90.64	94.98	93.42	94.70	95.13	89.57	92.13	93.93
ALIP	**97.48**	97.58	**98.01**	**95.16**	**95.78**	**96.41**	**94.96**	**95.32**	**95.62**

表4 ALIP与LPMF的运算时间对比

Table 4 Comparison of computation time between ALIP and LPMF

算法	运算时间/s		
	Citeseer数据集	DBLP数据集	Cora数据集
LPMF	60.892 8	53.695 1	36.401 8
ALIP	62.107 9	54.730 0	37.543 4

3.5 消融实验

为了证明输入层模块、矩阵分解模块以及高阶向量优化模块可提高节点之间链接的可能性,本文在Citeseer数据集、DBLP数据集、Cora数据集上进行消融实验,设置训练比率为0.7,0.8,0.9,表示向量长度为100,实验结果如图4所示。

在图4中,ALIP-IN算法为消融输入层模块的算法;ALIP-SVDS算法为消融矩阵分解的算法;ALIP-OP算法为消融向量优化的算法。

由图4可知,①在Citeseer数据集、DBLP数据集和Cora数据集上,ALIP算法*AUC*均高于ALIP-IN算法,并且有11.45百分点至28.18百分点的提升;②相较于ALIP-SVDS算法,ALIP算法*AUC*均高于ALIP-SVDS算法(Citeseer数据集除外),并且有0.9百分点至4.08百分点的提升;③在Citeseer数据集、DBLP数据集和Cora数据集上,ALIP算法*AUC*均高于ALIP-OP算法,并且有1.7百分点至3.42百分点的提升。综上所述,可有效证明在算法中加入输入层模块、矩阵分解模块以及向量优化模块可使得节点之间的链路预测能力增加,对ALIP算法有一定的帮助,进一步验证ALIP算法的可行性。

3.6 调参分析

在本文调参实验中,设置了训练比率T和向量长度K,$T=0.70,0.75,0.80,0.85,0.90,0.95$,$K=25,50,100,150,200,300$。调参分析结果如图5所示。

由图5可知,2个参数对链路预测算法性能均存在影响。当向量长度$K=200$、训练比率$T=0.95$时,在Citeseer数据集上,该算法的性能最好,其*AUC*为98.98%。在Citeseer数据集上,除了训练比率T为0.7~0.8且向量长度K为100,150,200,300以外,其余K值对应的*AUC*变化较大。在DBLP数据集上,不同向量长度对应的*AUC*的变化幅度较稳定,这是因为DBLP是稠密网络。在Cora数据集上,随着训练率的不断变化,不同向量长度对应的*AUC*总体趋势均呈上升状态并且波动较大,这是因为Cora数据集为稀疏网络。综上所述,训练比率T和向量长度K对稀疏网络影响较大,对稠密网络影响较小。

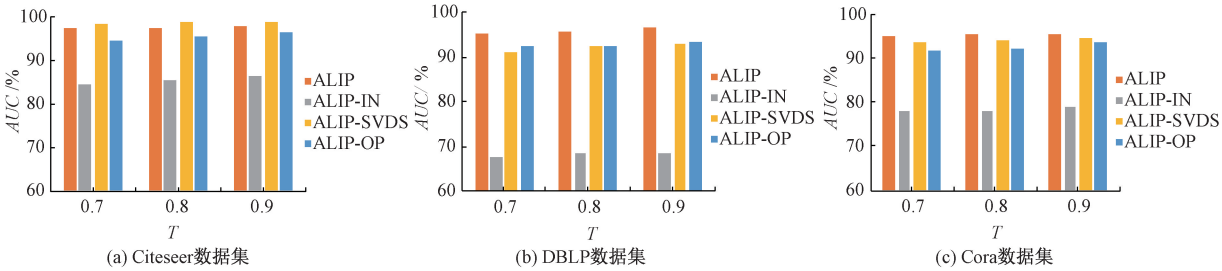

图4 消融实验结果

Figure 4 Ablation experiment results

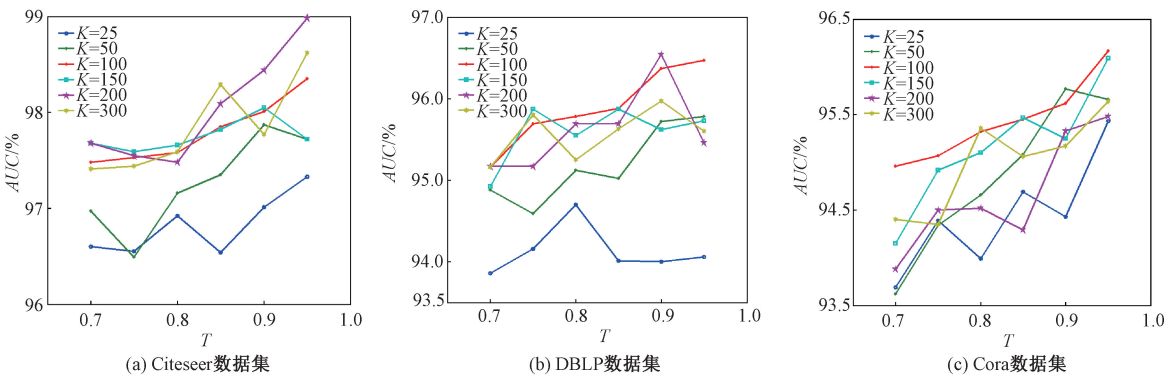

图5 调参分析结果

Figure 5 Tuning parameters analysis results

4　结论

针对无标签网络中基于图神经网络的链路预测方法使用其高效建模机制进行链路预测任务时性能较差的问题,提出了一种近似图神经网络框架的无监督链路预测算法。首先,输入层类比 GCN 的输入层,输入网络的结构信息和节点属性,确保网络特征的完整性;其次,进行矩阵分解和向量优化,得出网络节点表示向量,该步骤替代 GCN 的梯度更新和迭代优化过程;最后,计算相似度矩阵,使用 *AUC* 指标评测 ALIP 算法性能。实验结果表明,ALIP 算法可以高效利用网络特征,提高节点之间关系的预测精度。

参考文献:

[1] XIE X Q, LI Y J, ZHANG Z Q, et al. A joint link prediction method for social network[C]//International Conference of Young Computer Scientists, Engineers and Educators. Cham: Springer, 2015: 56-64.

[2] SCHAFER J L, GRAHAM J W. Missing data: our view of the state of the art[J]. Psychological Methods, 2002, 7(2): 147-177.

[3] KOSSINETS G. Effects of missing data in social networks [J]. Social Networks, 2006, 28(3): 247-268.

[4] LI M, MENG X M, ZHENG R Q, 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.

[5] DZAFERAGIC M, KAMINSKI N, MCBRIDE N, et al. A functional complexity framework for the analysis of telecommunication networks[J]. Journal of Complex Networks, 2018, 6(6): 971-988.

[6] 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.

[7] LYU L Y, ZHOU T. Link prediction in complex networks: a survey[J]. Physica A: Statistical Mechanics and Its Applications, 2011, 390(6): 1150-1170.

[8] ZHOU T, LYU L Y, ZHANG Y C. Predicting missing links via local information[J]. The European Physical Journal B, 2009, 71(4): 623-630.

[9] JACCARD P. Etude de la distribution florale dans une portion des Alpes et du Jura[J]. Bulletin De La Societe Vaudoise Des Sciences Naturelles, 1901, 37(142): 547 -579.

[10] SØRENSEN T, SØRENSEN T, BIERING-SØRENSEN T, et al. A method of establishing group of equal amplitude in plant sociobiology based on similarity of species content and its application to analyses of the vegetation on Danish commons[J]. Biologiske Skrifter, 1948, 5(4): 1-34.

[11] 王富田, 张鹏,肖井华. 链路预测算法错边识别能力的评测[EB/OL]. (2015-12-30)[2023-05-15]. http://www.paper.edu.cn/releasepaper/content/201512-1363.
WAMG F T, ZHANG P, XIAO J H. Evaluation the ability of link prediction methods in the spurious link detection [EB/OL]. (2015-12-30)[2023-05-15]. http://www.paper.edu.cn/releasepaper/content/201512-1363.

[12] KATZ L. A new status index derived from sociometric analysis[J]. Psychometrika, 1953, 18(1): 39-43.

[13] LEICHT E A, HOLME P, NEWMAN M E J. Vertex similarity in networks[J]. Physical Review E, Statistical, Nonlinear, and Soft Matter Physics, 2006, 73 (2): 026120.

[14] FOUSS F, PIROTTE A, RENDERS J M, et al. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2007, 19(3): 355-369.

[15] KOVÁCS I A, LUCK K, SPIROHN K, et al. Network-based prediction of protein interactions[J]. Nature Communications, 2019, 10: 1240.

[16] 廖亮, 张恒锋. 基于支持向量机的机会网络链路预测[J]. 信息通信, 2018, 31(9): 23-25.
LIAO L, ZHANG H F. Link prediction of opportunistic network based on support vector machine[J]. Information & Communications, 2018, 31(9): 23-25.

[17] 杨妮亚, 彭涛, 刘露. 基于聚类和决策树的链路预测方法[J]. 计算机研究与发展, 2017, 54(8): 1795 -1803.
YANG N Y, PENG T, LIU L. Link prediction method based on clustering and decision tree[J]. Journal of Computer Research and Development, 2017, 54(8): 1795-1803.

[18] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[EB/OL]. (2017-02-22)[2023-05-11]. https://arxiv.org/abs/1609.02907.

[19] LEI K, QIN M, BAI B, et al. GCN-GAN: a non-linear temporal link prediction model for weighted dynamic networks[C]//IEEE INFOCOM 2019 - IEEE Conference on Computer Communications. Piscataway: IEEE, 2019: 388-396.

[20] LI G H, MÜLLER M, THABET A, et al. DeepGCNs: can GCNs go as deep as CNNs? [C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE, 2019: 9266-9275.

[21] CHEN M, WEI Z W, HUANG Z F, et al. Simple and deep graph convolutional networks[EB/OL](2020-07-04)[2023-05-11]. https://arxiv.org/abs/2007.02133.

[22] YANG C, SUN M S, LIU Z Y, et al. Fast network embedding enhancement via high order proximity approximation[C]//Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence. Melbourne: International Joint Conferences on Artificial Intelligence Organization, 2017: 3894-3900.

[23] LIU Z, ZHANG Q M, LYU L Y, et al. Link prediction in complex networks: a local naïve Bayes model[J]. EPL (Europhysics Letters), 2011, 96(4): 48007.

[24] KLEIN D J, RANDI? M. Resistance distance[J]. Journal of Mathematical Chemistry, 1993, 12(1): 81-95.

[25] CHEBOTAREV P, SHAMIS E. The matrix-forest theorem and measuring relations in small social groups[EB/OL]. (2006-02-04)[2023-06-01]. https://arxiv.org/abs/math/0602070.

[26] 冶忠林, 曹蓉, 赵海兴, 等. 基于矩阵分解的DeepWalk链路预测算法[J]. 计算机应用研究, 2020, 37(2): 424-429, 442.
YE Z L, CAO R, ZHAO H X, et al. Link prediction based on matrix factorization for DeepWalk[J]. Application Research of Computers, 2020, 37(2): 424-429, 442.

[27] 曹蓉, 赵海兴, 冶忠林. 基于网络节点文本增强的链路预测算法[J]. 计算机应用与软件, 2019, 36(3): 227-235, 242.
CAO R, ZHAO H X, YE Z L. Link prediction algorithm based on text enhanced of network nodes[J]. Computer Applications and Software, 2019, 36(3): 227-235, 242.

[28] 王曙燕, 巩婧怡. 融合节点标签与强弱关系的链路预测算法[J]. 计算机工程与应用, 2022, 58(18): 71-77.
WANG S Y, GONG J Y. Link prediction algorithm fusing node label and strength relationship[J]. Computer Engineering and Applications, 2022, 58(18): 71-77.

An Unsupervised Link Prediction Algorithm Based on an Approximate Graph Neural Network Framework

LI Gege[1,2], YE Zhonglin[1,2], CAO Shujuan[1,2], ZHOU Lin[1,2], WANG Xueli[1,2]

(1. College of Computer, Qinghai Normal University, Xining 810008, China; 2. The State Key Laboratory of Tibetan Intelligent Information Processing and Application, Qinghai Normal University, Xining 810008, China)

Abstract: For unlabeled networks, the link prediction method based on graph neural networks had poor performance when using its efficient modeling mechanism for link prediction tasks. An unsupervised link prediction algorithm (ALIP) was proposed. It could approximate the graph neural network framework to simulate the efficient modeling mechanism and learning process of graph neural network algorithms, and to solve the problem of insufficient modeling caused by missing network node labels. Firstly, referring to the input layer of GCN, the structural information and node attributes of the network were fused. Secondly, matrix factorization is used to replace the hidden layer of GCN and simulate forward propagation. Then the ideas of identity mapping and vector optimization to achieve vector transformation and model optimization to obtain the network node representation vector, which were used to simulate the back propagation of GCN. Finally, the similarity matrix for performance evaluation of link prediction tasks was calculated. On the Citeseer dataset, DBLP dataset and Cora dataset, the experimental results showed that ALIP algorithm had a maximum *AUC* value of 98.01%, and its performance was superior to the other 23 link prediction algorithms. The effectiveness and feasibility of the algorithm, in this study provide a new solution for complex unlabeled network link prediction tasks.

Keywords: matrix factorization; vector optimization; graph convolutional neural network; similarity matrix; link prediction; high-order neighbors