DOI: 10.11992/tis.201806007
网络出版地址: http://kns.cnki.net/kcms/detail/23.1538.tp.20190109.1422.004.html

基于图勾勒的图链路预测方法

尤洁[1], 李劲[1,2], 张赛[1], 李婷[1]
(1. 云南大学 软件学院, 云南 昆明 650091; 2. 云南省软件工程重点实验室, 云南 昆明 650091)

摘 要: 针对已有链路预测算法复杂度高, 不适于在大规模图上进行链接预测的问题, 本文基于图勾勒近似技术对已有链路预测方法进行优化, 提出了基于图勾勒的链路预测方法。该方法将链路预测算法的计算复杂度由 $O(n^3)$ 降低至 $O(n^2k^2\log^2 n)$。为进一步提高链接预测效率, 给出了基于 Spark 的并行化链路预测实现方法。在真实图数据集上进行测试, 实验结果表明本文方法在保证链接预测精度的前提下, 可有效提升算法效率。

关键词: 图数据; 算法复杂度; 链路预测; 图勾勒; 节点相似性; 并行计算; Apache Spark
中图分类号: TP311 **文献标志码**: A **文章编号**: 1673−4785(2019)04−0761−08

中文引用格式: 尤洁, 李劲, 张赛, 等. 基于图勾勒的图链路预测方法 [J]. 智能系统学报, 2019, 14(4): 761–768.
英文引用格式: YOU Jie, LI Jin, ZHANG Sai, et al. Graph sketches-based link prediction over graph data[J]. CAAI transactions on intelligent systems, 2019, 14(4): 761–768.

Graph sketches-based link prediction over graph data

YOU Jie[1], LI Jin[1,2], ZHANG Sai[1], LI Ting[1]
(1. School of Software, Yunnan University, Kunming 650091, China; 2. Key Laboratory in Software Engineering of Yunnan Province, Kunming 650091, China)

Abstract: The high computational complexity of existing link prediction algorithms makes them unsuitable for link prediction on large-scale graphs. To solve this problem, we propose a novel link prediction approach that involves combining the existing link prediction approaches with graph sketch approximation. Our proposed approach reduces the computation complexity of link prediction from $O\left(n^3\right)$ to $O\left(n^2k^2\log^2 n\right)$ Furthermore, to enhance the efficiency of our approach; we also provide a parallel link prediction algorithm, which is implemented on the parallel computing framework Apache Spark. Finally, we conducted extensive experiments on a real network dataset to test the validation and efficiency of our approach. The experimental results indicate that our methods can effectively improve the efficiency of link prediction while guaranteeing prediction accuracy as well.

Keywords: graph data; algorithm complexity; link-prediction; graph sketches; nodes similarity; parallel computing; Apache Spark

图上的链路预测是指通过已有的网络拓扑结构和节点属性信息等预测网络中尚未产生连边的两个节点之间产生链接的可能性, 或者是已经产生但是并未发现的链接信息, 是图数据挖掘的重要方向之一, 受到广泛的关注。

当前, 关于链路预测的研究方法主要包括 3 种: 1) 基于极大似然估计的方法。该方法将网络链接看作是内在层次的反映, 采用极大似然估计进行预测。但该方法的预测准确性与样本数据量有关, 高质量的预测需要大的样本数据, 导致计算复杂度高, 不适用于大规模网络[1-2]; 2) 基于概率模型方法。通过建立可调参数模型再现网络的结构和关系特征, 将预测问题转化为预测边的属性问题进行预测, 此类方法具有较高预测精度, 但预测过程中涉及到非普适性的参数和节点

收稿日期: 2018−06−02. **网络出版日期**: 2019−01−10.
基金项目: 国家自然科学基金项目 (61562091); 云南省应用基础研究计划面上项目 (2016FB110).
通信作者: 李劲. E-mail: lijin@ynu.edu.cn.

属性信息，使得应用范围受限，计算复杂度高[3]；3) 基于节点相似性预测方法[4-14]。假设节点之间存在链接的可能性与节点之间的相似性紧密相关，通过预测节点之间的相似性来进行链路预测。其中，基于节点相似性模型的预测方法由于方法简单，链接预测质量较好等成为目前主流的链接预测方法。

但是，基于节点相似性的预测方法，其计算复杂度为 $O(n^3)$，在大规模图数据上进行链接预测时，算法执行效率低。为有效处理大规模网络，文献 [15] 提出了基于节点局部信息的分布式并行计算的预测方法。然而，该方法没有从降低时间复杂度的角度解决链路预测问题。

针对已有研究工作的不足，本文在保证链路预测质量的前提下，降低预测算法的计算复杂性角度，提出基于图勾勒[16] 的链路预测算法。首先，基于图勾勒技术对现有的链路预测方法进行扩展，定义了基于 ADS(all-distances sketches) 结构的链路预测相似性度量指标，提出了基于图勾勒的链路预测算法，将一般链路预测算法的计算复杂度由 $O(n^3)$ 降低至 $O(n^2k^2\log^2 N)$，其中 k 是 ADS 勾勒参数，n 是网络节点数。其次，基于并行图计算平台 Spark，提出了 ADS 的并行计算方法以及基于 ADS 技术的并行链路预测实现方法。从算法运算时间和预测精度两方面验证算法的有效性，实验结果表明：基于 ADS 技术的链路预测算法可以保证一定预测精度，同时降低预测方法的时间复杂度，提升运算效率。

1　背景知识

1.1　链路预测

给定无向图 $G=(V,E)$，其中，V 是顶点集，E 边集。$N=|V|$ 为网络节点数，$M=|E|$ 为边数。G 共有 $n(n-1)/2$ 个可能的节点对，所有节点对构成全集 U。令 $\overline{E}$ 表示 U 中不存在连边的节点对集合，且没有连边的节点对表示为 $(x,y)\in\overline{E}=U\backslash E$，其中 $x,y\in V$。给定一种链路预测方法，S_{xy} 表示节点对 (x,y) 的链路预测值，S 值越高，表示 (x,y) 出现连边的概率越高。

下面分别给出本文中采用的 3 种节点间相似度度量指标及定义：

定义 1　**Common Neighbor**(CN)[5]：如果图中两个节点拥有的共同邻居节点越多，那么这两个节点就越相似，则它们之间存在或者未来发生链接的可能性就越大。相似度定义为

$$S_{xy}^{\mathrm{CN}}=|\Gamma(x)\cap\Gamma(y)| \tag{1}$$

式中：x、y 表示节点；$\Gamma(x)$ 表示节点 x 的邻居节点集；$\Gamma(y)$ 表示节点 y 的邻居节点集；S 表示 x、y 的相似度的值；$|\Gamma(x)\cap\Gamma(y)|$ 表示节点 x 和节点 y 的共同邻居节点数。

定义 2　**Adamic Adar**(AA)[6]：AA 在 CN 的基础上，赋予邻居节点权重，它认为共同邻居节点的节点度对相似度也有影响，共同邻居节点度越大，它对节点相似度的贡献越小，反之，共同邻居节点度越小，它对节点的相似度的贡献越大。因此在求相似度的公式中，对共同邻居节点度赋予一个惩罚因子。其相似度定义为

$$S_{xy}^{\mathrm{AA}}=\sum_{z\in\Gamma(x)\cap\Gamma(y)}\frac{1}{\log(d_z)} \tag{2}$$

式中：x、y 表示图节点；S 表示 x、y 的相似度的值；$\Gamma(x)$ 表示节点 x 的邻居节点集；$\Gamma(y)$ 表示节点 y 的邻居节点集；z 表示 x、y 的共同邻居节点；d_z 表示 z 的节点度。

定义 3　**Resource Allocation**(RA)[7]：RA 从资源分配的角度考虑节点相似性。它认为没有直接相连的两个节点，资源可以从一个节点传递到另一个节点，它们的共同邻居节点是两个节点传递资源的媒介，每一个媒介都有一个单位的资源，它将自己的资源平均分配给它的邻居节点，另一个节点接收到的资源数就是这两个节点的相似度。其相似度定义为

$$S_{xy}^{\mathrm{RA}}=\sum_{z\in\Gamma(x)\cap\Gamma(y)}\frac{1}{d_z} \tag{3}$$

评估指标：链路预测结果的衡量指标主要包括 Precision(准确率)[17] 和 AUC(曲线下面积)[18]，Precision 针对局部结果进行评估，AUC 基于全局进行评估，本文讨论的是整体性能，故以 AUC 作为预测精度的评估标准。AUC 的值越高，则链路预测整体性能较好。

定义 4　对于边集进行数据划分，有 $E=E_T\cup E_p, E_T\cap E_p=\phi$，假设 $\bar{E}$ 是属于全集 U，但是不属于边集 E，从 E_p 中取出一条边的预测值记为 a，从 $\bar{E}$ 中选出一条边的预测值记为 b，比较 n 次，若 $a>b, n'=1$，若 $a=b, n''=1$，否则不计数，具体如下：

$$\mathrm{AUC}=\frac{n'+0.5n''}{n} \tag{4}$$

1.2　图勾勒技术

ADS(all-distances sketches) 是定义在图节点上的数据摘要结构。通过对图中各节点的可达邻居节点集进行抽样，抽样结果与原节点的集合构成

了该节点的 Sketch 结构。在大图上, 基于 ADS 可有效进行节点相似关系, 中心度等度量计算[16]。

定义 5　节点 v 的 All-Distances Sketches (ADS) 的定义如下[16]:

$$\mathrm{ADS}(v)=\{(u,d(v,u))|r(u)<K_r^{\mathrm{th}}(N(v,u))\} \tag{5}$$

式中: v 是 V 任意顶点; $r(u)$ 表示节点 v 的随机 rank 值, 即函数 $r:V\to[0,1]$ (对任意顶点 $v\in V, r(v)\sim U[0,1]$); $d(v,u)$ 表示从节点 v 到节点 u 的距离; $N(v,u)$ 表示比节点 u 更接近节点 v 的节点的集合 (即 $N(v,u)=\{i\in V|d(v,i)<d(v,u)\}$); $K_r^{\mathrm{th}}(X)$ 集合 X 中所有元素按照 rank 值从小到大排序, 第 K 个元素的 rank 值, 当 $K>|X|$ 时, $K_r^{\mathrm{th}}(X)=1$, K 值是平衡 sketch 规模大小和勾勒精度的参数。

例 1　图的 ADS 结构如图 1 所示, 该图为有向图带权图。节点上的数值为勾勒 ADS 结构所对应生成的 0 ~ 1 的随机数。

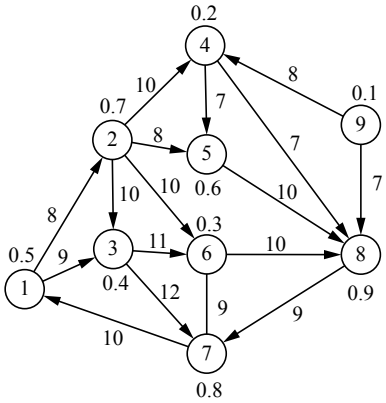


图 1　图的 ADS 示例

Fig. 1　An illustration of ADS in a graph

图中每个节点的 ADS 结构是一个集合。以节点 1 为例, ADS(1)($K=2$)={(1, 0), (2, 8), (3, 9), (4, 18), (6, 18)}表示在图中随机值取值情况下, ADS 勾勒参数为 2 时节点 1 的 ADS 结构, 集合中元素 (4, 18) 表示节点 1 到节点 4 的最短距离是 18。例如:

$$\mathrm{ADS}(1)=\{(1,0),(2,8),(3,9),(4,18),(6,18)\}$$

$$\mathrm{ADS}(2)=\{(2,0),(5,8),(4,10),(6,10),(3,10)\}$$

$$\mathrm{ADS}(5)=\{(5,0),(8,10),(7,19),(6,28),(1,29),(3,38),(4,47)\}$$

从给出的 ADS(1)($K=2$) 可以看出, 每个节点与其 ADS 集合里面对应的节点是可达关系, 但是每个节点的 ADS 集合里面并没有包含所有的可达节点, 只包含了部分可达节点, 在 ADS 中包含多少可达节点与勾勒参数 K 取值的大小相关, K 取值越大, 勾勒的精度越高, ADS 的尺寸越大。

2　链路预测方法

ADS 是对节点的全局邻居节点进行抽样, 而 CN、AA、RA 3 种算法的默认情况是基于 1 跳邻居进行计算的, 故为了排除多跳邻居对相似度的影响, 基于节点的 ADS 结构的链路预测算法中也只考虑一跳邻居节点。基于 1 跳邻居的 ADS 的大小永远不大于节点的 1 跳邻居数, 所以在求两个集合的相似度时, 运算量也相应减少。在 AA 算法和 RA 算法中还涉及到求共同邻居节点的度, 其他相似性度量指标也涉及到节点中心度的计算等, 这个过程中需要耗费大量的计算时间, 而 ADS 抽样的过程中会过滤掉一部分的邻居节点, 故在一定程度上减少了部分求节点度、中心度的运算量。

对图勾勒后, 得到的 ADS 结构不再是单一的节点集、边集所构成的图数据, 而是由节点及其部分邻居节点构成的集合, 这部邻居节点包括了一跳至多跳另据节点, 还带有相应的可达距离, 故 ADS 需要根据自身结构定义合适的相似性指标, 具体定义如下:

定义 6　基于 ADS 的 CN 度量指标 (ADS-CN) 定义如下:

$$S_{xy}^{\mathrm{ADS-CN}}=\left|\mathrm{ADS}(x)_1\cap\mathrm{ADS}(y)_1\right| \tag{6}$$

式中: x、y 表示待求相似度的节点; S 表示 x、y 的相似度的值; $\mathrm{ADS}(x)_1$ 表示节点 x 的 ADS 概要结构并且可达节点集的距离 $x\leqslant 1$; $\mathrm{ADS}(y)_1$ 表示节点 y 的 ADS 概要节点且可达节点距离 $y\leqslant 1$; $\left|\mathrm{ADS}(x)_1\cap\mathrm{ADS}(y)_1\right|$ 表示节点 x 和节点 y 基于 ADS 的概要结构的一跳共同邻居数。

定义 7　基于 ADS 的 AA 度量指标 (ADS-AA) 如下:

$$S_{xy}^{\mathrm{ADS-AA}}=\sum_{z\in\mathrm{ADS}(x)_1\cap\mathrm{ADS}(y)_1}\frac{1}{\log(d_z)} \tag{7}$$

式中: d_z 表示节点 z 在图中的节点度, 其余符号定义同定义 6。

定义 8　基于 ADS 技术扩展的 RA 度量指标 (ADS-RA) 定义如下:

$$S_{xy}^{\mathrm{RA}}=\sum_{z\in\mathrm{ADS}(x)_1\cap\mathrm{ADS}(y)_1}\frac{1}{d_z} \tag{8}$$

公式中符号含义同定义 6。

2.1　基于 ADS 勾勒技术的链路预测算法

首先简要介绍链路预测算法的基本思想, 链路预测算法首先将待预测数据集 E 划分为训练集 E_T 和测试集 E_p。找出训练集中不存在连边的节点对, 得到 E 中不存在连边的数据集 $\bar{E}$ 和 E_p, 并计算节点对的相似度值, 随机从 $\bar{E}$ 和 E_p 中各选出一条边, 比较它们的相似度的值, 重复多次, 根据 AUC 公式定义, 得到预测精度。基于 ADS

勾勒技术的链路预测算法的基本思想：在计算节点对相似度之前，构造出边集 E_T 的图结构，对图进行 ADS 勾勒处理，得到 E_T 中每个节点的 ADS 结构，根据基于 ADS 结构定义的相似性度量指标进行链路预测。由于节点的 ADS 是独立于图的，这样带来的优势是原图有些节点发生变化以后，只需要更新变化节点的 ADS，带来的好处是可以独立动态更新节点的 ADS 结构，更新代价小；处理后的数据另一个优点是利于并行化处理，每个节点及其 ADS 结构与其他节点时独立的，在其他并行框架下，每个节点 ADS 互不干扰，利于并行。

基于 ADS 勾勒技术的链路预测算法的具体描述如算法 1。

分析算法 1 的时间复杂度。首先，由文献 [16] 可知，对于图中的一个节点 v，ADS(v) 的期望大小为 $K+K(H(n_v)-H(K))\approx K(1+\log n_v-\log K)$ 其中 n_v 是从节点 v 出发的可达邻居节点数，$H(i)=\sum_{j=1}^{i}\frac{1}{j}$ 是第 i 个调和级数，由于 $n_v\leqslant n$ ($|V|=n$) 且 $H(n)=O(K\log n)$，所以 ADS(v) 的期望大小为 $O(K\log n)$。于是，基于 ADS 技术求图中节点相似度的时间复杂度为 $O(n^2K^2\log^2 n)$。

算法 1　基于 ADS 勾勒技术的链路预测算法

输入　$G(V,E)$, 预测值比较次数 n, 勾勒参数 K;

输出　AUC 值。

1) 切割边集 E 为训练集 E_r 和测试集 E_p，$E=E_r\cup E_p, E_r\cap E_p=\varnothing$;

2) degree$(v)\leftarrow$ 求出 E_r 中所有结点的结点度;

$(G'=(V,E_r), v\in V)$

3) ADS$(v)\leftarrow$ 根据式 (3) 构造图 G' 中个结点的;

ADS$(G'=(V,E_r), v\in V)$

//找出训练集中不存在连边的结点对集合

4) $A\leftarrow\{(v,w)|v,w\in V$且$(v,w)\in\overline{E_r}\}$;

5) $S_{vw}\leftarrow$ 求出 A 中所有节点对的预测值;

//得到 $\overline{E}$ 中所有连边结点对的预测值;

6) $S_{xy}\leftarrow\{S(x,y)|x,y\in V$且$(x,y)\in\overline{E}\}$;

//得到 E_p 中所有连边结点对的预测值;

7) $S_{uv}\leftarrow\{S(u,v)|u,v\in V$且$(u,v)\in E_p\}$;

8) for $i\leftarrow 1$ to n do

9) a: 从 S_{uv} 中选出一条边的预测值;

10) $b: S_{xy}$ 中选出一条边的预测值;

11) n': 表示 E_p 中的预测值大于 $\overline{E}$ 中预测值的次数;

12) n'': 表示 E_p 中的预测值等于 $\overline{E}$ 中预测值的次数;

13) if $a>b$ do

14) $n'=n'+1$;

2.2　基于 ADS 勾勒技术的并行化链路预测算法

为提高链路预测算法的执行效率，在算法 1 基础上，进一步提出了基于 Spark 的并行化的链路预测算法。该算法具体描述如算法 2。算法 2 的执行过程与算法 1 一致，但算法 2 将算法 1 中的每一步骤采用弹性分布式数据集 (RDD) 进行了实现。基于 RDD 表示，采用对 RDD 的 Map-reduce 并行化操作有效提升链接预测算法的执行效率。RDD 转换和操作细节详见算法 2 中的描述。

算法 2　基于 Spark 的并行化链路预测算法

输入　$G(V,E)$, 预测值比较次数 n;

输出　AUC 值。

//创建边集 E 的 RDD

1) dataRDD ← sc.textFile(edgesPath);

//创建训练集 E_r 的 RDD

2) edgestRDD ← dataRDD.takeSample (0.9 * (dataRDD.count));

//创建测试集 E_p 的 RDD

3) edgespRDD ← dataRDD.subtract(edgestRDD);

//找出训练集中不存在连边的结点对集合

4) pairRDD ← edgestRDD.cartesian(edgestRDD).subtract(edgestRDD);

//求出各顶点的邻居节点

5) verticesRDD ← edgestRDD.groupByKey() .map (vertices.nbrs);

6) g ← Graph(verticesRDD, pairRDD); //构造图 g

//求出各结点的节点度

(7) degreeRDD ← verticesRDD.map(nbrs.size);

//求结点 x, y 的相似度

8) simRDD ← g.triplets.map(sim(x, y.persist()));

9) simepRDD ← 从 simRDD 筛选 E_p 连边的预测值;

10) simeNotRDD ← 从 simRDD 筛选 E_P 连边预测值;

11) a ← simepRDD.takeSample(true, n);

3　实验结果

3.1　实验环境设置

表 1 给出了本文实验数据集统计信息。其中，N 表示网络中节点的总数，E 表示网络中边的总数，$\langle ad\rangle$ 表示网络节点的平均度，$\langle d\rangle$ 表示网络的平均最短距离，C 表示网络的集聚系数。

表 1　实验数据集拓扑结构信息

Table 1　Experimental dataset topology information

数据集	*N*	*E*	<*ad*>	<*d*>	*C*
USAir97	332	2 126	12.807	2.46	0.749
Yeast	2 361	6 646	5.63	4.38	0.388
PB	1 224	19 090	27.36	2.51	0.361

本文实验在 USAir97(美国航空网络数据[17])、Yeast(酵母菌蛋白质相互作用网络数据[18])、Grid(美国电力网络数据[19])3 个数据集上进行测试, 实验结果主要对链路预测算法和基于 ADS 勾勒技术的链路预测算法两种算法的运行时间和预测精度进行对比分析。实验环境包括内存: 64 GB; 处理器: inter(R) Xeon(R) CPU E5-2620 v3 @ 2.40 GHz 2.40 GHz; 开发平台: Intellij IDEA 2016.2.5+ Spark GraphX; 开发语言: Scala。

本次所有实验结果均是对数据集进行 10 次划分, 求平均值, 由于程序运行时间存在误差, 故每次划分结果得到训练集在运行相关算法的程序时, 运行 20 次求平均值作为划分一次的数据值。AUC 计算公式中的 n 统一取 10 万次。

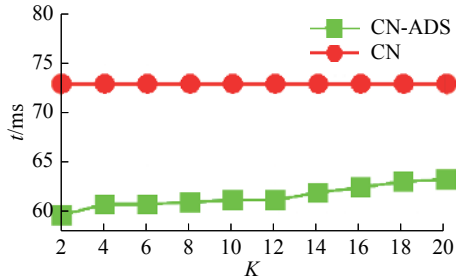
3.2　基于 ADS 的链路预测算法的有效性

ADS 勾勒技术是对原数据的一种抽样方法, 通过抽样达到降低计算复杂度的目的, 但是由于它只是对数据的近似勾勒, 所以用勾勒的结果进行数据的分析与挖掘, 在精度上会有一定的损失, 是不可避免的, 但是损失一定范围内的精度, 却提升了较大的计算效率。

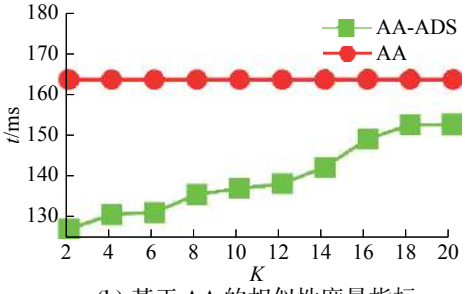
实验结果已经表明, 基于 ADS 的链路预测方法在 K 取较优值时, 精度损失的范围远远小于计算效率提升的范围, 所以得出 ADS 技术在链路预测算法中对降低算法时间复杂度, 提升计算效率是有效的。

3.2.1　两种算法执行效率

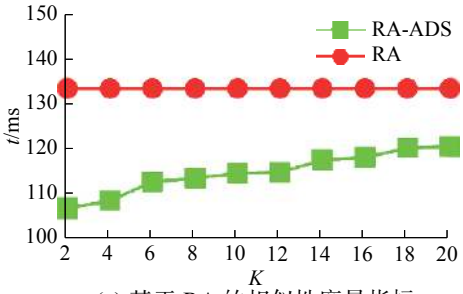
图 2~4 分别给出了 USAir97、Yeast、Grid 数据集基于 CN、AA、RA3 种相似性度量指标的两种算法的执行效率。从图中可以看出基于 ADS 勾勒技术的链路预测算法执行时间均低于原链路预测算法的执行时间, 由于链路预测算法不涉及到 K 值得变化, 故在 K 值变化过程中结果不改变。而基于图勾勒技术的链路预测算法随着 K 值的变化算法执行时间有所增加, 但是均低于原链路预测算法, 计算效率提高了约百分之 15%~25%, 这是由于 ADS 结构是原数据集的一个抽样, 每个节点的一跳邻居节点集的数目远远小于原图的一跳邻居节点集的数目, 当 K 值足够大时, 抽样的结果也只能等于原图的数据。



(a) 基于 CN 的相似性度量指标



(b) 基于 AA 的相似性度量指标



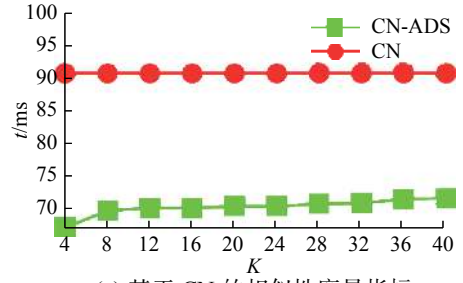
(c) 基于 RA 的相似性度量指标

图 2　CN、AA、RA 度量指标运行时间对比 (USAir97)

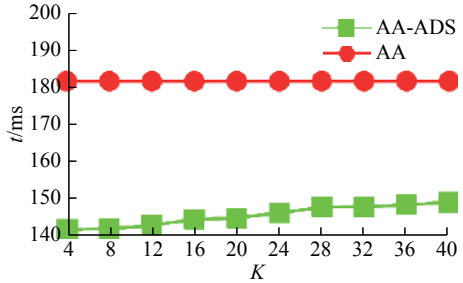
Fig. 2　CN, AA, RA metrics comparison of run time (USAir97)

3.2.2　两种算法的预测精度

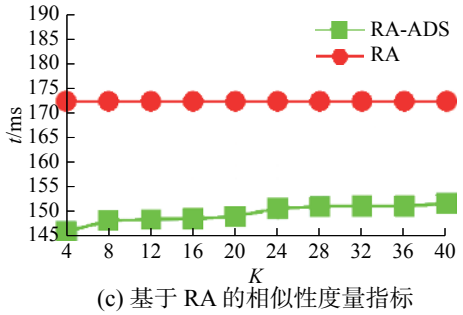
图 5~7 给出了 3 个数据集在两种算法下的预测精度, 实验结果显示, 基于 ADS 的链路预测算法的预测精度随着 K 值的增加而逐渐接近于原链路预测算法的精度, 数据线最后趋于重合。



(a) 基于 CN 的相似性度量指标



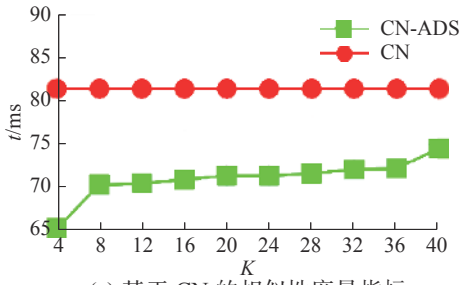
(b) 基于 AA 的相似性度量指标



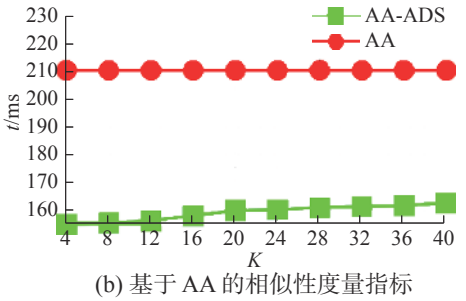
(c) 基于 RA 的相似性度量指标

图 3　CN、AA、RA 度量指标运行时间对比 (Yeast)

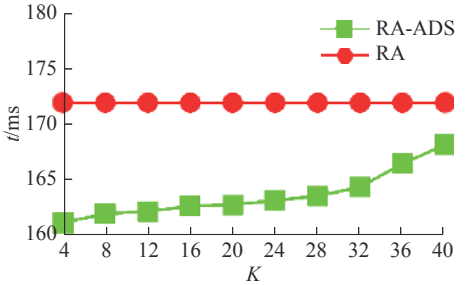
Fig. 3　CN，AA，RA metrics comparison of run time (Yeast)



(a) 基于 CN 的相似性度量指标



(b) 基于 AA 的相似性度量指标



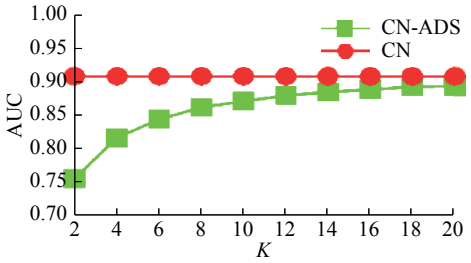
(c) 基于 RA 的相似性度量指标

图 4　CN、AA、RA 度量指标运行时间对比 (Grid)

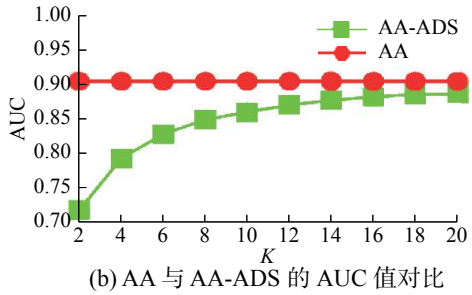
Fig. 4　CN，AA，RA metrics comparison of run time (Grid)

从表 1 中可以看出 USAir97 数据集节点远小于 Yeast 数据集和 Grid 数据集，但是图中结果显示 USAir97 数据集较为理想的预测结果对应的 K 值要比其余两个数据集对应的 K 值要大，这是由于 USAir97 数据集要比 Yeast 数据集和 Grid 数据集稠密，在网络刻画中对精度的要求更高，所以相对而言预测结果较为理想的情况下对应的 K 值要大。

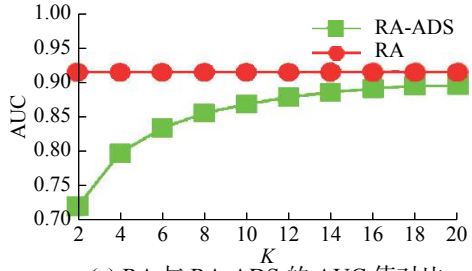
图 5 和图 6 中精度的变化逐渐上升最后趋于稳定，但是图 7 中精度的变化有波动，在千分之一上下波动，存在原因可能有两个：1) 计算 AUC 过程中抽取的次数不够所造成的误差；2)ADS 节点随机值变化过程中产生的误差。



(a) CN 与 CN-ADS 的 AUC 值对比



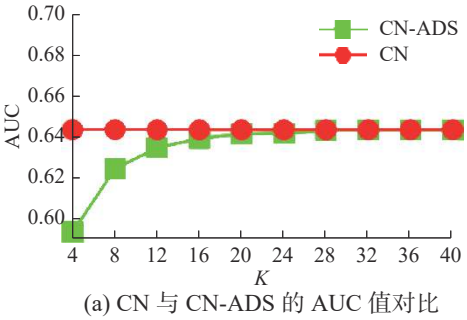
(b) AA 与 AA-ADS 的 AUC 值对比



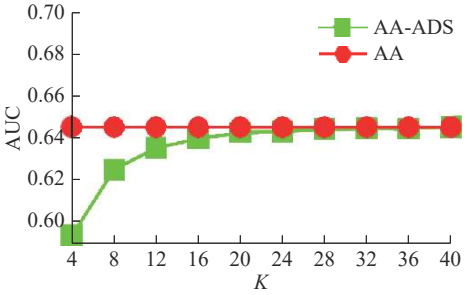
(c) RA 与 RA-ADS 的 AUC 值对比

图 5　CN，AA，RA 度量指标 AUC 对比 (USAir97)

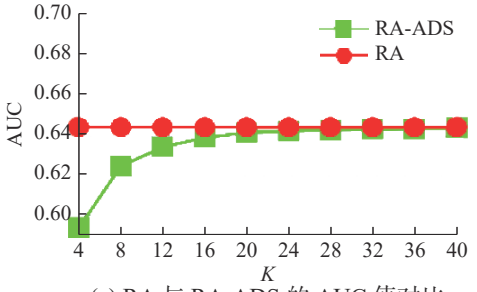
Fig. 5　Comparison of the CN, AA, RA metrics AUC (USAir97)



(a) CN 与 CN-ADS 的 AUC 值对比



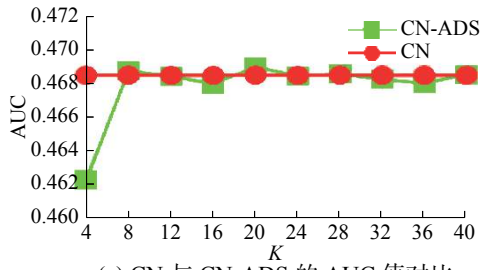
(b) AA 与 AA-ADS 的 AUC 值对比



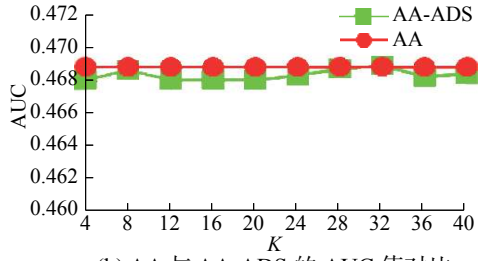
(c) RA 与 RA-ADS 的 AUC 值对比

图 6　CN、AA、RA 度量指标 AUC 对比 (Yeast)

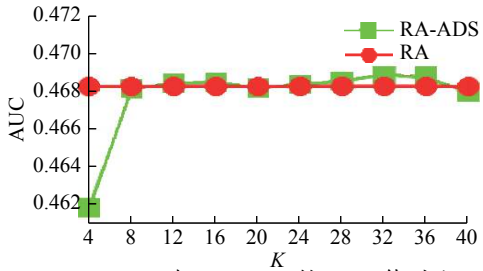
Fig. 6　Comparison of the CN，AA，RA metrics AUC (Yeast)



(a) CN 与 CN-ADS 的 AUC 值对比



(b) AA 与 AA-ADS 的 AUC 值对比



(c) RA 与 RA-ADS 的 AUC 值对比

图 7　CN、AA、RA 度量指标 AUC 对比 (Grid)

Fig. 7　comparison of the CN，AA，RA metrics AUC (Grid)

3.3　基于 ADS 与基于网嵌入的链路预测算法对比

DeepWalk[20] 是一种基于随机游动的网络表示学习方法。通过 DeepWalk 可获得图中节点的向量化表示，进而可基于向量点积进行链接预测。在真实图数据上将本文方法与基于 DeepWalk 的链接预测方法进行了实验对比。测试数据为蛋白质交互网络[21](protein-Protein Interactions)。该数据包括 19 706 个节点、390 633 条边。采用 CN-ADS 与 DeepWalk 在算法执行时间和 AUC 值上进行了比较。其中 DeepWalk 的参数设置为：向量学习模型为 Skip-Gram，向量维数设为 64。实验结果如图 8、9 所示。从图 8、9 结果可知，小 K 值可保证算法执行效率，然而，AUC 较 DeepWalk 差。提高 K 值后，在执行时间仍小于 DeepWalk 的情况下，可显著改善 AUC 值。特别地，当 K >32 后，AUC 值优于 DeepWalk。对于链接预测而言，本文算法在一定条件下优于 DeepWalk 的结果。

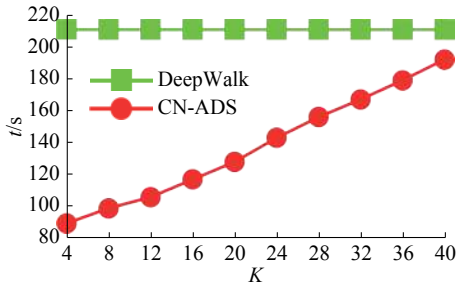


图 8　PPI 数据集上 CN-ADS 与 DeepWalk 的时间对比

Fig. 8　Time comparison of CN-ADS and DeepWalk on PPI

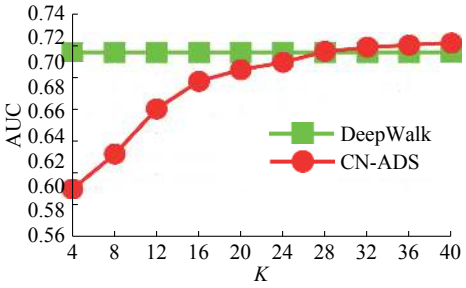


图 9　PPI 数据集上 CN-ADS 与 DeepWalk 的 AUC 对比

Fig. 9　AUC comparison of CN-ADS and DeepWalk on PPI

4　结束语

本文针对大规模网络数据在链路预测中存在时间复杂度高、运算量大等问题，对现有的链路预测方法进行扩展，结合现有的图勾勒技术，提出了基于 ADS 技术的链路预测方法，根据勾勒的结果结合现有的预测方法，定义了基于 ADS 结构的链路预测方法，在算法预测精度和预测时间中取得了较好的折衷，并在真实网络数据中验证了算法的有效性。

本文是基于局部信息相似度进行链路预测的，更精确的预测方法是基于全局信息进行预测的，如何更好地在图勾勒技术的基础上基于全局信息定义预测方法，是将来展开的要点之一。此外，为验证图勾勒技术在链路预测问题上面的有效性，本文是通过实验数据进行验证分析的，缺少严谨的理论证明，后续工作将会致力于从理论方面证明图勾勒技术对链路预测的有效性。

参考文献：

[1] 吕琳媛. 复杂网络链路预测 [J]. 电子科技大学学报, 2010, 39(5): 651–661.
LYU Linyuan. Link prediction on complex networks[J]. Journal of University of Electronic Science and Technology of China, 2010, 39(5): 651–661.

[2] CLAUSET A, MOORE C, NEWMAN M E J. Hierarchical structure and the prediction of missing links in networks[J]. Nature, 2008, 453(7191): 98–101.

[3] AIROLDI E M, BLEI D M, FIENBERG S E, et al. Mixed membership stochastic blockmodels[J]. Journal of machine learning research, 2008, 9: 1981–2014.

[4] YU Kai, CHU Wei, YU Shipeng, et al. Stochastic relational models for discriminative link prediction[C]//Proceedings of the 19th International Conference on Neural Information Processing Systems. Vancouver, Canada, 2006: 1553–1560.

[5] LIBEN-NOWELL D, KLEINBERG J. The link—prediction problem for social networks[J]. Journal of the association for information science and technology, 2007, 58(7): 1019–1031.

[6] ADAMIC L A, ADAR E. Friends and neighbors on the Web[J]. Social networks, 2003, 25(3): 211–230.

[7] LYU Linyuan, JIN Cihang, ZHOU Tao. Similarity index based on local paths for link prediction of complex networks[J]. Physical review E, 2009, 80(4): 046122.

[8] ZHOU Tao, Lü Linyuan, ZHANG Yicheng. Predicting missing links via local information[J]. The European physical journal B, 2009, 71(4): 623–630.

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

[10] LEICHT E A, HOLME P, NEWMAN M E J. Vertex similarity in networks[J]. Physical review E, 2006, 73: 026120.

[11] KLEIN D J, RANDIĆ M. Resistance distance[J]. Journal of mathematical chemistry, 1993, 12(1): 81–95.

[12] 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.

[13] BRIN S, PAGE L. The anatomy of a large-scale hypertextual Web search engine[J]. Computer networks and ISDN systems, 1998, 30(1-7): 107–117.

[14] JEH G, WIDOM J. SimRank: a measure of structural-context similarity[C]//Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Edmonton, Alberta, Canada, 2002: 538–543.

[15] 饶君, 吴斌, 东昱晓. MapReduce 环境下的并行复杂网络链路预测 [J]. 软件学报, 2012, 23(12): 3175–3186.
RAO Jun, WU Bin, DONG Yuxiao. Parallel link prediction in complex network using mapreduce[J]. Journal of software, 2012, 23(12): 3175–3186.

[16] COHEN E. All-distances sketches[M]//KAO M Y. Encyclopedia of Algorithms. New York: Springer, 2016: 2320-2334.

[17] BATAGELJ V, MRVAR A. Pajek datasets[EB/OL]. 2006 http://vlado.fmf.uni-lj.si/pub/networks/data/default.html.

[18] BU Dongbo, ZHAO Yi, CAI Lun, et al. Topological structure analysis of the protein–protein interaction network in budding yeast[J]. Nucleic acids research, 2003, 31(9): 2443-2450.

[19] WATTS D J, STROGATZ S H. Collective dynamics of ‘small-world’ networks[J]. Nature, 1998, 393(6684): 440–442.

[20] PEROZZI B, AL-RFOU R, SKIENA S. DeepWalk: online learning of social representations[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2014: 701-710

[21] BREITKREUTZ B J, STARK C, REGULY T, et al. The bioGRID interaction database: 2008 update[J]. Nucleic acids research, 2008, 36(S1): D637-D640.

作者简介：

尤洁，女，1991 年生，硕士研究生，主要研究方向为数据与知识工程。

李劲，男，1975 年生，副教授，中国人工智能学会不确定性人工智能专委会委员，主要研究方向为数据与知识工程。

张赛，女，1994 年生，硕士研究生，主要研究方向为数据与知识工程。