基于子图演化与改进蚁群优化算法的社交网络链路预测方法

顾秋阳[1,2,3]，琚春华[4]，吴功兴[4]

（1. 浙江工业大学管理学院，浙江 杭州 310023；2. 浙江工业大学中国中小企业研究院，浙江 杭州 310023；
3. 宁波诺丁汉大学商学院，浙江 宁波 315175；4. 浙江工商大学管理工程与电子商务学院，浙江 杭州 310018）

摘 要：基于改进蚁群优化算法与子图演化，提出了一种新型非监督社交网络链路预测（SE-ACO）方法。该方法首先在社交网络图中确定特殊子图；然后研究子图演化以预测图中的新链接，并用蚁群优化算法定位特殊子图；最后针对所提方法使用不同网络拓扑环境与数据集进行检验。结果表明，与其他无监督社交网络预测算法相比，所提 SE-ACO 方法在多数数据集上的评估结果较好，且运行时间较短，这表明图形结构在链路预测算法中起重要作用。

关键词：链路预测；蚁群优化算法；社交网络；子图演化

中图分类号：TP391

文献标识码：A

doi: 10.11959/j.issn.1000−436x.2020223

Social network link prediction method based on subgraph evolution and improved ant colony optimization algorithm

GU Qiuyang[1,2,3], JU Chunhua[4], WU Gongxing[4]

1. School of Management, Zhejiang University of Technology, Hangzhou 310023, China
2. China Institute for Small and Medium Enterprises, Zhejiang University of Technology, Hangzhou 310023, China
3. Business School, University of Nottingham Ningbo, Ningbo 315175, China
4. School of Management Science & Engineering, Zhejiang Gongshang University, Hangzhou 310018, China

Abstract: Based on improved ant colony algorithm and subgraph evolution fusion, a new unsupervised social network link prediction method (SE-ACO) was proposed. First, the special subgraph was determined in the social network graph. Then the evolution of the subgraph was studied to predict the new links in the graph, and the special subgraph was located by the ant colony method. Finally, using different network topology environments and data sets to test the proposed method. Compared with other unsupervised social network prediction algorithms, the proposed SE-ACO method has the best evaluation results, shorter running time and the best effect on most data sets, which indicates that graph structure plays an important role in link prediction algorithm.

Key words: link prediction, ant colony optimization algorithm, social network, subgraph evolution

1 引言

近年来在线社交网络规模和用户数量日益增加，链路预测作为社交网络分析中的一个新兴课题，现有研究往往根据社交网络图中现有节点及链接属性来预测 2 个节点之间的新链接[1]。链路预测的目标为通过考虑社交网络图在时刻 t 的快照，预测出该网络图在时刻 t+1 可能产生新链接，而时刻 t+1 可能是拍下快照后的一周、一个月、一年，甚至几年[2]。Liben-Nowell 等[3]将社交网络中的链路预测视为挖掘链接的子集。链路预测可用于许多不同的应用中。王智强等[4]等认为在许多网络中，由于

收稿日期：2020−06−09；**修回日期**：2020−09−05

基金项目：国家自然科学基金资助项目（No.71571162）；浙江省社科规划重点课题基金资助项目（No.20NDJC10Z）；浙江省自然科学基金资助项目（No.LQ20G010002）

Foundation Items: The National Natural Science Foundation of China (No.71571162), Zhejiang Social Science Planning Key Projects Fundation (No.20NDJC10Z), The Natural Science Foundation of Zhejiang Province (No.LQ20G010002)

信息或数据带有噪声，会产生一些不必要的链接，可以借助于链路预测方法来检测这些不必要的链接，可用于通信侦察领域。其通过对社交网络进行链路预测得到了网络的演化模型，使研究者能更好地了解网络。Huang 等[5]提出了基于链路预测算法的新型似然方法，使用社交网络图对传染病的患病率进行建模，每个预测链接均显示了社会中潜在的感染区域。Yin 等[6]在社交网络环境中，通过预测用户中的共同兴趣来发掘新好友（即新链接），此类推荐也会增加用户对社交网络的忠诚度。Pech 等[7]通过分析客户购物情况进行链路预测，形成客户商品推荐，优化推荐系统，提升了预测成功率。

如今常用的链路预测方法可分为两大类，即有监督式方法和无监督式方法。在预测链接时不需要任何先验知识或培训的为无监督式方法，而使用该网络训练所得的模型及先验知识来预测链接的为有监督式方法。无监督式方法大多使用网络相似度度量及结构属性来实现链路预测，且不需要进行任何训练。Wang 等[2]认为两节点之间最短路径的长度和公共邻节点的数量都可视为结构属性，这些属性均可以用于链路预测。Jaccard 等[8]提出的 Jaccard 系数等方法类似于公共邻点，而不同之处在于，在这种相似度度量方法中，如果两节点所拥有的公共邻点较多、非公共邻点较少，则两节点越靠近彼此。节点度数也是社交网络环境中预测新链接的关键结构属性之一，即节点度数较高的 2 个节点以后彼此交互的可能性更大，这种方法又被称为优先链接[9-11]。而近年来，某些链路预测算法所用的方法是基于特殊子图的演化，此部分将在第 2 节进行详细介绍。最大似然方法为典型的有监督链路预测算法。Wu 等[12]介绍了一种由网络数据推导的层次结构似然估计方法（系统树图），而概率模型也可视为有监督式链路预测算法。王凯等[13]利用二元分类器进行链路预测。虽然有监督式方法考虑了每种网络的特殊性，但可能会耗费大量时间，且不适用于大型网络[14]。

蚁群优化（ACO, ant colony optimization）算法最早由 Dorigo 等[15]提出，旨在解决困难的组合问题。蚁群优化算法以蚂蚁的觅食行为为基础，认为蚂蚁在觅食时会随机搜寻周围环境；在发现食物后，蚂蚁会检查食物的数量和质量并取走一块食物；在回家的路上，蚂蚁会沿途释放一种叫作信息素的化学物质，信息素的数量与食物源的数量和质量成正比。信息素即为蚁群间的协同机制，可让它们找出食物源到家的最短路径[15]。本文拟使用蚁群优化算法对所研究的问题进行建模，主要包括以下 3 个步骤。

步骤 1 设蚂蚁 k 由社交网络图中节点 i 遍历至节点 j 的概率为 P_{ij}^k，如式(1)所示。

$$P_{ij}^k = \frac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\sum_{L\in N_i^k}\left(\tau_{iL}^{\alpha}\eta_{iL}^{\beta}\right)} \tag{1}$$

其中，τ_{ij} 为边信息素数量，η_{ij} 为由节点 i 移动至节点 j 的成本，N_i 为节点 i 的邻节点集，系数 α 和 β 分别为蚁群优化算法的基本参数。本文假设所有边都具有初始数量的信息素。

步骤 2 无论何时使蚂蚁都能找到食物为该问题的理想解，且每只蚂蚁在从食物源到家的沿途上都会释放信息素，如式(2)和式(3)所示。

$$\Delta\tau_{ij}^k = \begin{cases} \dfrac{1}{c^k}, & (i,j)\in T^K \\ 0, & \text{其他} \end{cases} \tag{2}$$

$$\tau_{ij} = \tau_{ij} + \sum_{k=1}^{m}\Delta\tau_{ij}^k \tag{3}$$

其中，T^K 为蚂蚁 k 所遍历的由家至食物源的路径，其长度为 c^k；m 为现有蚁群的总数，表示找出由食物源至家的最短路径。本文假设每条边上的信息素都会逐渐消退；路径越拥挤，其上保留的信息素就越多。

步骤 3 分析信息素的消退机理，如式(4)所示。

$$\tau_{ij} = (1-\rho)\tau_{ij} \tag{4}$$

其中，ρ 为基本参数。

本文首先以子图演化的链路预测为基础，在社交网络图中确定特殊子图。然后，研究子图演化以预测图中的新链接，并用蚁群优化算法来定位特殊子图。最后，本文针对所提方法使用不同的网络拓扑环境与数据集进行实验，以期为社交网络环境的优化、链路的预测、监控与追责提供实验依据；同时也为大众更好地使用社交网络提供指导意见。

本文的主要贡献包括：1) 提出了一种基于子图演化与蚁群优化算法融合的社交网络链路预测方法，提升了预测精度；2) 证明了图形结构在链路预测算法中起到重要作用；3) 相对于图神经网络及图深度学习等热门链路预测方法，SE-ACO 算法能够有效地缩短运算时间。

2　文献回顾

2.1　文献综述

由于本文所提方法运用蚁群优化算法找出了特殊子图，因此本文将介绍基于子图查找和演化的链路预测方法。本文所提方法与其他方法的最大区别在于将蚁群优化算法首次用于链路预测过程中。Fadaee 等[16]认为通过找出一定时间间隔内的三元组演化模型，可预测出 2 个连续社交网络快照间的新链接，其所述方法为有监督式结构链路预测算法。如果该图为有向图，则会有 64 个三元组，在每个三元组中的每 2 个节点之间有 4 种不同的关系：一个双向链接，2 个单向链接，一个无链接。计算 2 个连续社交网络快照上的 64 个三元组可得到三元组转换矩阵（TTM, triad transition matrix），基于此可找出 2 个非连接节点之间的链接概率。Lichtenwalter 等[17]提出了一种新型有监督式社交网络链路预测及分析算法，旨在找出社交网络图的子结构，也称为顶点配置结构（VCP, vertex collocation profile）。张子柯等[18]将聚类系数的定义进行扩展，提出了一种新型概率链路预测算法。聚类系数是社交网络图中的三元组变为三角关系的趋势，式(5)展示了如何计算社交网络图中的聚类系数。

$$\text{Clustering Coeffcient}=\frac{3\text{Triang}}{\text{Triads}} \tag{5}$$

Zhang 等[19]研究了有向网络中的特殊子图，认为这些子图代表有向网络中的微观组织原理，并表明，这些子图中在社交网络中较普遍。最优的有向网络局部结构是包括 4 个节点与 4 个有向链接的 Bi-Fan 结构，胡文斌等[20]基于聚类机制（CM, clustering mechanism）和潜在理论（PT, potential theory）证实了上述观点，即如果子图是仅比 Bi-Fan 结构少一个链接的子图，则创建该链接的概率将是最高的。郭丽媛等[21]采用谱聚类（SC, spectral clustering）的方法对社交网络的链路预测进行研究，通过降维方法获得简易矩阵, 采用该矩阵能够提高预测效果；其不仅计算了类内节点之间的相似度，还计算了不同类之间的节点相似度。Gong 等[22]首先提出了一个特征转化方法，该方法能够把原有特征进行简化，提高了链路预测精度；其次构建基于受限玻尔兹曼机的深度学习算法进行链路预测。这种无监督算法仅使用小样本训练即可取得良好效果。

目前在社区发现和预测这一领域，图神经网络（GNN, graph neural network）和图搜索聚类算法也表现出了较优能力。Gori 等[23]借鉴神经网络的研究成果，设计了一种用于处理图结构数据的模型。Bronstein 等[24]对图结构数据和流形数据领域的深度学习方法进行了综述，侧重于将所述各种方法置于一个称为几何深度学习的统一框架内。白铂等[25]提出了一种新的图神经网络分类方法，重点介绍了图卷积网络，并总结了图神经网络方法在不同学习任务中的开源代码和基准。Fan 等[26]将图神经网络提取的物品特征和节点特征，通过多层感知机进行评分预测推荐。郭嘉琰等[27]指出了在不同 GNN 变体中出现的相关聚集过程，但也发现了以图神经网络为代表的方法通常时间成本较高。

2.2　文献评述

通过对现有研究成果的梳理发现，社交网络中的链路预测方法已经受到了国内外学者的重视并得到丰富的成果。上述研究成果对本文具有一定的借鉴意义，但也存在一些不足：1)已有很多学者对基于社交网络图的链路预测开展了一系列卓有成效的研究，但是多数将其定义为静态图（如李冬等[28]），鲜有基于子图演化进行动态社交网络链路预测的文献；2) 现有的社交网络链路预测方法大多基于数理推导（如 Gao 等[11]），但很少有加入如蚁群优化算法等概率型路径寻优算法进行社交网络链路预测的文献；3) 几乎所有的相关文献都使用了例如 BA 无标度网络（BA scale-free network）、WS 小世界网络（WS small world network）等人工网络进行实验（如方哲等[29]），很少有使用真实数据集对社交网络链路预测进行分析的文献记录；4) 几乎所有的相关文献都直接使用现有算法进行链路预测（如尚凤军等[30]），很少有使用改进算法对社交网络进行链路预测的文献。

基于以上综述，本文首先提出一种基于子图演化与蚁群优化算法融合的社交网络链路预测方法，以基于子图演化的链路预测为基础，在社交网络图中确定特殊子图；然后研究子图演化以预测图中的新链接，并用蚁群优化算法定位特殊子图；最后针对所提方法使用不同网络拓扑环境与数据集进行验证。

3　社交网络链路预测方法设计

3.1　方法设计总体思路

社交网络中的每个实体都可以用一个节点表

示，2 个节点之间的关系可用链接表示。为预测 2 个节点之间可能产生的新链接，那么这 2 个节点之间至少应该有一个关联。另一方面，本文认为预测 2 个无任何关系的节点之间的链接为一项随机任务，特别是对于大型社交网络而言，该问题至关重要，这是由于人们可以从这些社交网络的预测列表中剔除许多候选链接。Lichtenwalter 等[17]认为这就是如果 2 个节点之间最多相距 2 个跳点，则大多数链路预测算法都认为这 2 个节点之间有潜在链接的原因。故本文认为 2 个节点链接的预测问题可以转化为预测社会群体中源节点和目标节点之间链接的问题。例如社会群体中的节点存在相似兴趣、利益或目标，则源节点与目标节点社会群体可能存在更多共同兴趣（即链接），则这 2 个节点之间创建新链接的概率较高。社交网络中最简单的社会群体为单节点，在单个关系的无向网络中，单节点社群的源节点和目标节点之间至多存在一种关系。要对这 2 个节点之间的链接进行预测，其之间就必须存在这种单一关系。为了对社交网络中的新链接进行预测，应考虑更复杂的社群，这些社群内部至少要存在 2 个节点，且随着社群中节点数量的增多及源节点与该社群间关系的增多，源节点与社群中目标节点之间创建新链接的概率也会增大。

单个关系无向网络中某节点和某社群间可能的相关性如图 1 所示。图 1 中第一层显示了节点和社群的最简单关系，该层中单个关系无向网络中仅存在一个链接，如时刻 t 的快照所示。时刻 t 的快照表示一种网络结构，链路预测算法用该结构预测下个快照中的新链接。如前所述，每 2 个实体间至少应有一个链接，这样才能预测下个快照中的链接。图 1 中的 2 个实体分别为左侧的源节点和右侧的目标社群。第一层中未保留链接来预测网络中的下一快照。第二层的目标社群由 2 个节点组成，该层中时刻 t 的快照表示源节点与社群间的关联，时刻 t+1 的快照表示可创建的可能链接（虚线）。由于所有链接和节点的权重相同，因此图 1 中未描绘其他可能的同构交互。第二层中有 4 种不同的关系，因此存在 4 种不同的可能链接。第四层是本文关注点的重点，其群体结构为三角关系，左右两部分分别有 2 个可能的链接。本文将讲述如何根据第四层的子图结构来预测链接。而第三层中为最常见的社交网络结构，在此不做赘述。图 1 所示四大层次并不是如今算力所能解决的，还可以考虑更多节点、更复杂的群体结构，但由于找出这些的群体成本更高，因此本文未考虑更复杂的层次，仅示例社交网络中两实体间更复杂的交互层次交互关系，如图 2 所示。

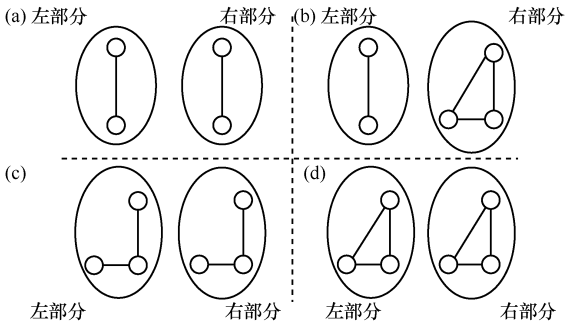


图 2　社交网络中两实体间更复杂的交互层次交互关系

图 1 中的第四层定义了本文所述方法，具有一个节点和一个三角关系，且三角关系中至少存在一条公共边。第四层有 2 个子图(a)和(b)。由于子图(b)

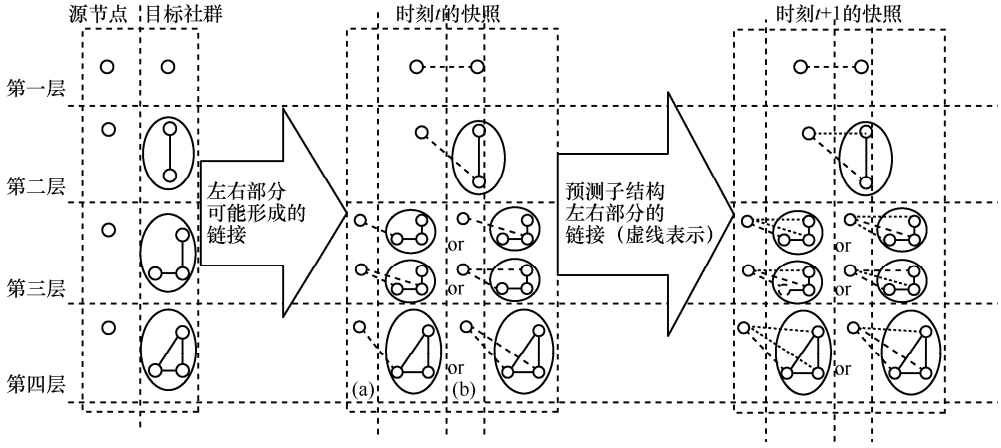


图 1　单个关系无向网络中节点与社群间可能存在的交互关系

的结构属性多于子图(a)，而结构属性越多，预测出正确链接的机会就越大，故子图(b)价值更大。为证明这一观点，本文对比了根据子图(a)与子图(b)所预测的链接的准确性。首先从社交网络样本中提取出子图(a)和子图(b)，然后考察了作为潜在链接的所有潜在链接（即图 1 中的虚线），最后根据精度度量对比了这 2 个子图所预测出的链接质量。对比结果显示本文观点是正确的，即如果使用更多的结构属性，可以更正确地预测链接。Huang[31]的实验也证实了该结论，具体遍历与循环数计算方式参考文献[31]。因此，根据黄璐等[32]提出的概率模型，子图(b)中仅有一条边不存在的概率要大于子图(a)中 2 个链接都不存在的概率。本文对图 1 的其他层也进行了同样的实验，并将其结果与第四层结果进行了对比，发现第四层中的子图结构更有利于社交网络情景中的链路预测，故本文用第四层的子图结构来进行社交网络链路预测。

算法 1 描述了 SE-ACO 算法的具体过程，其目的是由社交网络在时刻 t 的快照找出图 1 中子图(a)与子图(b)，然后预测这些子图中在时刻 t+1 的快照的潜在链接（即图 1 中的虚线链接）。接下来，根据蚁群优化算法找出的社交网络中的三角关系。找到三角关系后，本文试图根据这些三角关系找出图 1 中的子图(a)与子图(b)。由于某些链接在多个子图结构中是共用的，故其评分会更高。最后，本文按评分降序排列得出预测链接的列表。

算法 1　SE-ACO 算法

输入　三角关系 triangles(G)

输出　预测链接 result ← result+link

require:　$G=(V,E)$　//`加载导入网络 (G)

1) procedure SE-ACO(G)　//在网络 (G) 中找到三角关系（见算法 2）

2)　load(G)　//列出预测值

3)　triangles ← find triangles(G)　//得到网络 (G) 中找到三角关系的数量

4)　result ← null

5)　n ← size (triangles)

6)　$i \leftarrow 1$

7)　while $i \leqslant n$ do

8)　newLinks ← predict (triangles)[i]　//基于每个三角关系进行链路预测（见算法 3）

9)　for all link ∈ newlinks do

10)　if result contains link then

11)　result[link]++ //如果之前的链接是被预测的，则将分数进行累加

12)　else

13)　result ← result+link　//将新的预测链接添加到最终的预测列表中

14)　end if

15)　end for

16)　$i \leftarrow i+1$

17)　end while

18)　result ←sort descending result　//将新的预测链接添加到最终的预测列表中

19)　return result

20)　end procedure

3.2　基于改进蚁群优化算法定位三角关系的模型构建

为了找出三角关系，本文对蚁群优化算法进行了改进，与原始蚁群优化算法的区别主要有以下几点。

1) 假设初始状态下蚂蚁没有家，这表示所有蚂蚁在一开始时是分散在社交网络的各个节点中的。

2) 与原始蚁群优化算法（如式(1)所示）相反，SE-ACO 算法中的蚂蚁更倾向于选择信息素含量较低的路径。这一特性使蚂蚁能够有更高的概率对社交网络图上未曾勘探过的部分进行探索。蚂蚁 k 由节点 i 移动至节点 j 的概率如式(6)所示，其中，τ 为对应边的信息素含量；α 为基本参数，本文设 α=1。与式(1)不同，即 SE-ACO 算法与传统蚁群优化算法的区别，式(6)令 β=0，且其两者图中的信息素含量成反比。如果给释放的信息素一个负值系数，则蚂蚁移动与信息素之间为正相关关系。

$$P_{ij}^{k}=\frac{\left(\frac{1}{\tau_{ij}}\right)^{\alpha}}{\sum_{L\in N_i^k}\left(\frac{1}{\tau_{ij}}\right)^{\alpha}} \tag{6}$$

3) 食物（即解）呈现为三角关系，而蚂蚁的目标是找出三角关系。由于每个三角关系中都存在 3 个节点，而蚂蚁拥有如图 3 所示的特殊记忆力，故蚂蚁在社交网络图上的每次移动均会记录在其记忆中。如果记忆存在已满，则将数据写入先前的记忆单元内（图 3 中数字表示蚂蚁循环覆盖的特殊记忆

力，箭头代表循环覆盖的方向），但首先要检验准备写入的数据是否等同于先前的记忆单元，如果相等，则认为找到了三角关系。

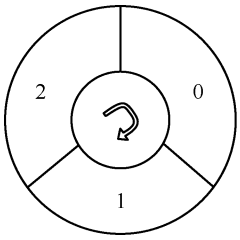


图3 蚁群优化算法中蚂蚁的记忆结构图

4) 令所有边上的信息素的初始数量均等于一个单位，如果信息素属于已找出的三角关系中的边，则边上的信息素将增加。每当发现一个三角关系时，该三角关系中所有边上的信息素将增加一个单位，SE-ACO 算法中找到食物（即解）的方法为找出社交网络图中的三角关系。

5) 各边所释放的信息素不会消退。

6) 蚂蚁有死亡属性。鉴于该特性，蚂蚁不会勘探社交网络图中已访问过的部分，整个社交网络图都已被充分勘探后，由于不需要做进一步勘探，故所有蚂蚁都会死亡。蚂蚁死亡的条件如下：之前已勘探过的任意边均不含有初始信息素；困在 2 个节点孤岛或边缘节点中；蚂蚁的周围环境都充满了信息素。

使用 SE-ACO 算法进行三角关系查找的具体过程如算法 2 所示。首先，创建一定数量的蚂蚁并将它们随机放置于社交网络节点中。由于蚂蚁会死亡，故蚂蚁的初始数量一般为社交网络图中节点数量的 10～20 倍，且在移动 2 次或 3 次后，会有将近一半的蚂蚁死亡，而当所有蚂蚁死亡或迭代次数超过特定次数后，算法停止。在大多数网络中，算法 2 的平均迭代次数为 10。每次迭代时，新节点被蚂蚁选中的概率如式(6)所示，同时检查网络中是否发现新的三角关系，如发现则将该三角关系中所有边的信息素增加一个单位，然后将该三角关系加入已找到的三角关系列表中。每次迭代时均检查蚂蚁的健康状况，如果符合死亡条件中的任一条件，则该蚂蚁死亡。

算法 2 SE-ACO 算法寻找三角关系

输入 初始化信息素 initialize pheromone(E)

输出 三角关系 triangles(G)

require: $G=(V,E)$ //需要网络(G)

1) procedure find triangles(G)

2) initialize pheromone(E) //在社交网络(G)的每条边中都加入初始化信息素

3) ants ←initialize ants(V) //初始化蚂蚁并其随机放到图节点中

4) (triangles) ←null //列出三角关系

5) antsNumber ←number of (ants) //得到蚂蚁的数量

6) iteration ←1

7) while 0<iteration ≤ Maxiteration do

8) for all ant ∈ ants do

9) next ←Choose Next Node() //根据概率（式(6)）选择下一节点

10) if privious node in memory()==next then //如果找到了三角关系

11) triangles ←save triangle()

12) increase pheromone of (triangles) //增加一个单位的三角边信息素

13) triangles ← triangles+triangle

14) else

15) Put Into Memory(next)

16) end if

17) if Check Health(ant)== false then //如果蚂蚁不健康则将其剔除

18) Delete(ant)

19) end if

20) end for

21) antsNumber ←number of (ants) //再次得到蚂蚁的数量

22) iteration++

23) end while

24) return triangles

25) end procedure

易知社交网络图中三角关系的时间复杂度为$O(n^3)$，其中，n为社交网络图中的节点数量。参照 Gong 等[22]的做法，查找三角关系最快的算法的时间复杂度为$O(n^{2.376})$，空间复杂度为$O(n^2)$。对稀疏图来说，常用方法的时间/空间复杂度较低，对低权值图来说，现有算法的时间复杂度应为$O(mn^{\frac{1}{\alpha}})$，空间复杂度为$O(n^2)$，其中，m为边的数量，α的取值范围为[2,3]。对于大型非稀疏图而言，这些算法的时间和空间利用率普遍不高。

Latany 等[33]在大型低权值网络中引入了一种新型三角计算算法，并使用 Tsourakakis 等[34]所提特征三角形算法进行三角关系计算。刘树新等[35]所述三角计算方法的应用之一为链路预测，其理念为社交网络中朋友的朋友也是朋友的概率较大，故能生成最多三角关系的链接即是链路预测的最佳候选链接，这一理念与 Newman 等[36]所提的公共邻点算法非常接近。由于链路预测不需要找出社交网络图中三角关系的确切数量，故 Newman 等[36]所提的公共邻点算法并未找出社交网络图中三角关系的确切数量。要在大型社交网络图中找出三角关系的确切数量非常耗时，且成本很高。本文将改进蚁群优化算法用于查找三角关系，且将其应用于 MapReduce 框架（一种高度并行的分布式大规模数据处理框架）中[37]，这提升了算法的可扩展性。如前所述，用于查找三角关系的所述方法的时间复杂度为 $O(n)$。边上的信息素也可用于图形分区，这意味着信息素含量较低的边为图形分区的最佳候选点。而边上所释放的信息素也可用于确定节点的中心性，节点所连接的具有较多信息素的边越多，该节点的中心性越强[38]。

3.3　基于三角关系进行预测链接

本文根据上文中找出的三角关系来进行预测链接（如算法 3 所示）。首先，确定已找出三角关系节点的所有邻点，然后检验每个邻居节点是否属于图 1 中子图(a)或子图(b)中的一种。由于子图(b)优于子图(a)，因此如果社交网络图不是稀疏的，仅考察子图(b)就足够了。如果找出了一定数量的子图(a)或(b)，则这些子图中不存在的链接将被视为潜在链接。对于每个预测链接，可根据三角关系节点边上的信息素数量及它们所属的子图类型来计算其分数。信息素越高，则分数越高，子图(b)中的链接的分数较高。预测链接在某些子图结构间可能是重叠的，且要预测多次。每次进行一次链路预测，其分数就越高。2 个子图(b)之间的重叠链接如图 4 所示，由于该链接会预测 2 次，故该链接的分数也较高。

算法 3　SE-ACO 算法的链路预测

Require　triangle　//输入三角关系

1) procedure predict (triangles)

2) neighbors ←neighbors (triangle)　//得到三角节点的所有邻节点

3) newlinks ← null //列出新的预测链接

4) for all　neighbor ∈ neighbors　do

5) if neighbor belong subgraph(b) then //如果相邻节点属于子图(b)（见图 1）

6) link ← Get Non-Existed Link (b) //得到子图(b)中不存在的链接

7) calculate score(link) //根据三角节点边上的信息素进行计算打分

8) newlinks ← newlinks+link

9) elseif neighbor belong subgraph(a) then　//如果相邻节点属于子图(a)（见图 1）

10) links ←get non-existed link (a)　//得到子图(a)中不存在的链接

11) calculate score(link)　//根据三角关系节点中边的信息素进行计算打分

12) newlinks ← newlinks+link

13) end if

14) end for

15) return newLinks

16) end procedure

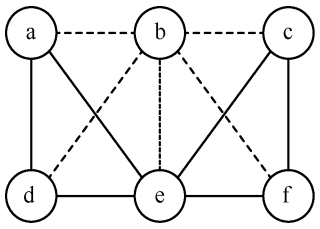


图 4　2 个子图(b)之间的重叠链接

4　实验验证与结果分析

首先对数据集及其特征进行介绍，然后将 SE-ACO 算法在这些数据集上的评估结果与其他非监督式结构链路预测算法进行比较。使用 Top-n 精度和接收器操作特性（ROC, receiver operating characteristic）曲线下面积及精确率−召回率曲线等方法进行评估，最后根据已执行的不同评估指标对算法结果进行了讨论。

4.1　数据说明

各数据集在时刻 t 和时刻 t+1 的统计信息分别如表 1 和表 2 所示。SE-ACO 算法使用了时刻 t 的数据集来预测时刻 t+1 的链接。其中节点和边的数量表示社交网络图中存在多少个节点和边，平均聚类系数表示三元组变成三角关系的倾向性度量，网络的聚类系数越高每个图中生成的链接越多。相称系数[39]是进行相似性度量的关键参数，度数相同的

表 1 各数据集在时刻 t 的统计信息

数据集	节点个数/个	边数/边	平均聚类系数	密度	同配性系数	SCC 中的节点个数/个	SCC 中的节点边数/边	估计直径	SCC	平均节点出度
新浪微博	37 864	593 832	67.40%	3.40%	54.90%	28 994	487 732	32	241	8.47
Twitter	49 822	709 223	59.10%	3.80%	38.10%	42 542	684 483	24	127	6.44
Facebook	38 942	493 801	72.40%	1.70%	39.40%	29 475	387 492	26	110	5.64
hep-ph	15 393	239 840	54.80%	0.40%	63.70%	13 874	203 342	14	173	9.83
astro-ph	19 831	493 208	65.00%	0.30%	49.20%	13 220	467 743	16	236	7.28
dblp-collab	389 403	2 783 964	59.40%	0.10%	39.10%	278 331	2 344 900	27	8 943	9.44
dblp-cite	26 495	398 504	14.30%	1.20%	−1.70%	11 048	284 931	8	5 639	8.28
polblogs	2 448	48 754	17.40%	2.40%	−3.50%	1227	35 622	6	374	12.09
patent-colla	609 444	5 890 483	60.40%	0.10%	27.10%	287 366	2 473 816	47	39 849	8.89

表 2 各数据集在时刻 t+1 的统计信息

数据集	加权 (yes/no)	控制 (yes/no)	节点个数/个	边数/边	节点增加数/个	边增加数/边	平均聚类系数	密度	同配性系数	SCC 中的节点个数/个	SCC 中的节点边数/边	估计直径	SCC	平均节点出度
新浪微博	no	no	42 093	617 898	4 229	24 066	68.30%	4.10%	52.70%	29 864	509 483	31	487	10.78
Twitter	no	no	55 891	790 937	6 069	81 714	60.50%	4.70%	39.10%	43 880	728 839	21	268	9.89
Facebook	no	no	42 965	528 908	4 023	35 107	71.10%	1.60%	38.70%	31 285	459 833	24	139	7.84
hep-ph	no	no	17 833	268 337	2 440	28 497	55.20%	0.50%	65.70%	14 087	220 983	13	340	13.09
astro-ph	no	no	20 931	509 838	1 100	16 630	63.00%	0.20%	50.40%	16 908	567 480	15	309	9.89
dblp-collab	yes	no	390 084	2 980 456	681	196 492	59.20%	0.10%	40.90%	289 844	3 192 086	25	4 059	14.58
dblp-cite	yes	yes	27 709	409 836	1 214	11 332	15.20%	1.00%	−2.00%	16 094	310 929	7	6 784	10.35
polblogs	no	yes	2 694	51 297	246	2 543	24.50%	2.60%	−4.60%	2 094	43 957	5	509	11.98
patent-colla	yes	no	687 943	5 984 760	78 499	94 277	62.20%	0.10%	23.60%	309 821	3 587 939	42	40 939	16.83

节点比其他节点更容易彼此关联，该度量范围为[−1,1]。相较于相称系数接近−1 的网络，在相称系数接近 1 的网络中度数相同的节点彼此的关联度最高。强连通分量（SCC, strongly connected component）表示社交网络图中的子图，由于大型网络的直径计算非常耗时，故本文使用 Lichtenwalter 等[40]中的近似算法来估算网络直径，使用五折交叉验证法来评估所提架构的性能。使用 Rapidminer 数据挖掘工具随机选取各用户评级数据的 20%作为测试集，并将剩余 80%的用户数据作为训练集。

本文使用国内外共 9 个相关数据集对 SE-ACO 算法进行验证，并利用 Python 工具从新浪微博、Twitter 与 Facebook 的 API（application programming interface）端口选取 10 个节点作为初始节点，爬取真实用户数据集作为实验仿真的基础数据，爬取时间为 2019 年 1 月 1 日—6 月 30 日，数据集分别包含 37 864 个新浪微博节点、49 822 个 Twitter 节点和 38 942 个 Facebook 节点，记为时刻 t；爬取时间为 2019 年 7 月 1 日—12 月 31 日，数据集分别包含 42 093 个新浪微博节点、55 891 个 Twitter 节点和 42 965 个 Facebook 节点，记为时刻 t+1。除此之外，本文认为如果科研论文作者的合作网络、网站之间的信息分享网络与专利合作网络等都应纳入考察范围。但由于数据的可得性问题，故本文使用以下数据集进行分析。hep-ph 与 astro-ph 表示科研论文作者合作网络，其中，hep-ph 为物理现象学领域，astro-ph 为天体物理学领域[2]。2004 年— 2006 年撰写论文的样本为时刻 t，2007 年—2009 年撰写的论文的样本为时刻 t+1。对上述 2 个数据集，本文仅使用核图，且使用撰写 3 篇论文及以上的作者作为节点。dblp-collab 和 dblp-cite 均来自于 DBLP 计算机科学文献[17]。其中，dblp-collab 为计算机科学作者合作网络，该网络中的时刻 t 为 2001 年— 2003 年的作者合作，快照时刻 t+1 为 2004 年—2005 年的作者合作；dblp-cite 表示相互引用的计算科学论文网络，该网络的时刻 t 为 1997 年—1998 年，时刻 t+1 为 1999 年—2000 年。Polblogs 为 2004 年的美国政治网络及网站间的链接[41]，将该网络图中的后 20%划分为

时刻 t+1，其余则为时刻 t。patent-colla 为节点为专利作者的数据集，其链接表示专利作者间的合作[42]。时刻 t 为 2006 年—2007 年作者的合作，时刻 t+1 为 2008 年—2009 年的合作，并将所有数据以 csv 格式保存在 MySQL 数据库中以便进行数据处理。对每一个社交网络数据集，采用 80%的数据进行训练，剩余的 20%用于测试。

4.2　实验结果

为体现 SE-ACO 算法的优越性，本文将其与 10 种不同的非监督式结构链路预测算法进行了对比。这 10 种算法的简要描述如下。

1) 公共邻点（CN, common neighbor）。令 $\Gamma(x)$ 表示节点 x 的邻点数，2 个领点具有的公共邻点越多，则链路预测任务[36]的分数越高，如式(7)所示。

$$\mathrm{Score}(x,y)=\left|\Gamma(x)\cap\Gamma(y)\right| \tag{7}$$

2) AA（Adamic-Adar）算法。Adamic 等[41]设计了一种用于检测 2 个网页相似度的度量，这种度量也可用于度量社交网络中两节点的相似度，具体如式(8)所示。其中，文献表明度数较小的 2 个节点的公共邻点比其他节点更有价值。

$$\mathrm{Score}(x,y)=\sum_{z\in\Gamma(x)\Gamma(y)}\frac{1}{\log\left|\Gamma(z)\right|} \tag{8}$$

3) Jaccard 系数算法。该相似度度量类似于寻找公共邻点，不同之处在于，就此度量方法而言，如果 2 个节点有较多的公共邻点和较少的非公共邻点，则它们的相似度较大[8]，如式(9)所示。

$$\mathrm{Score}(x,y)=\frac{\left|\Gamma(x)\cap\Gamma(y)\right|}{\left|\Gamma(x)\cup\Gamma(y)\right|} \tag{9}$$

4) 优先连接（PA, preferential attachment）算法。节点度数是预测新链接的关键属性。度数较高的 2 个节点在未来彼此交互的可能性越大[9-11]，这一理论可由式(10)推导得出。

$$\mathrm{Score}(x,y)=\left|\Gamma(x)\right|\left|\Gamma(y)\right| \tag{10}$$

5) Katz（Katz 指数）算法。该度量根据两节点之间的路径数及其长度来确定节点之间的相似度[43]，如式(11)所示。

$$\mathrm{Score}(x,y)=\sum_{L=1}^{\infty}\beta^{L}\left|\mathrm{paths}_{x,y}^{L}\right| \tag{11}$$

其中，$\mathrm{paths}_{x,y}^{L}$ 为节点 x 至节点 y、且长度为 L 的所有路径。系数 $\beta>0$ 用于降低长路径对分数的影响。在本文仅考虑 $\beta=0.005$，且路径的最大长度为 5。

6) Distance（距离）算法。使用距离算法进行相似度度量时，距离越近的两节点关联的机会越高，因此，距离 2 个跳点的节点，其彼此关联的概率最大[3]。在该度量算法中，距源节点具有相同数量跳点的节点与源节点形成的链接都具有相同的打分。

7) RP（Rooted PageRank）算法。该算法基于随机游走，由 PageRank[44]算法改写而成，并被文献[2]用于链路预测，如式(12)所示。其中，$H_{x,y}$（节点 x,y 的首次接触时间）为随机游走者由节点 x 移动至节点 y 所需的步数。由于击中时间不对称，因此 $H_{y,x}$ 的首次接触时间也是不同的。式(12)中的 π_x 与 π_y 均为平稳概率，为防止随机游走者距离起始节点 x 太远，这里使用了概率 α，即随机游走者返回至起始节点的概率为 α，移动至邻点的概率为 $1-\alpha$。本文实验设 $\alpha=0.5$。

$$\mathrm{Score}(x,y)=-(H_{x,y}\pi_y+H_{y,x}\pi_x) \tag{12}$$

8) SR（SimRank）算法。在该相似度度量算法中，2 个节点越相似，则分数越高。该算法的主要思想是根据社交网络的趋同性[11]进行预测。SimRank 算法[45]可由式(13)～式(15)表示，如果 2 个节点与较多的相似节点有关联，则这 2 个结果较相似。式(14)中的 γ 取值为[0, 1]，本文实验中，设 $\gamma=0.8$。

$$\mathrm{Score}(x,y)=\mathrm{Similarity}(x,y) \tag{13}$$

$$\mathrm{Similarity}(x,y)=\gamma\frac{\sum_{a\in\Gamma(x)}\sum_{b\in\Gamma(y)}\mathrm{Similarity}(a,b)}{\left|\Gamma(x)\right|\left|\Gamma(y)\right|} \tag{14}$$

$$\mathrm{Similarity}(x,x)=1 \tag{15}$$

9) PF（PropFlow）算法[46]。该算法为一种限制性随机游走预测器，是横向优先搜索的变形，本文实验中仅考虑了最大长度为 5 的路径。

10) 资源分配（RA, resource allocation）算法。该算法是基于复杂网络中的资源分析理念提出的[4]。其中，节点 x 可借助于邻点向节点 y 发送资源。简化情况下，每个节点仅向目标节点发送一个单位的资源，且该资源将发送至该节点的所有邻节点。节点 x 和 y 的相似度为它从节点 x 所得到的资源数量。参考文献[4]中的做法，使用式(16)中的 $\deg(z)$ 表示节点 z 的度数。

$$\mathrm{Score}(x,y)=\sum_{z\in\Gamma(x)\cap\Gamma(y)}\frac{1}{\deg(z)} \tag{16}$$

本文在 Python 环境中分别使用 Scipy[47]、Numpy[48]和 LPmade 工具包[40]执行上述算法。用于计算链接精度和召回率的真正例（TP, true positive）、真负例（TN, true negative）、假正例（FP, false positive）和假负例（FN, false negative）的定义如表 3 所示。而召回率（Recall）、精确率（Precision）、真正率（TPR, true positive rate）和假正率（FPR, false positive rate）分别如式(17)～式(20)所示。

表 3 链路预测中 TP、TN、FP 和 FN 的定义

参数	定义
TP	正确预测链接的数量
TN	正确的未预测链接的数量
FP	错误预测链接的数量
FN	错误的未预测链接的数量

$$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}} \tag{17}$$

$$\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}} \tag{18}$$

$$\text{TPR}=\frac{\text{TP}}{\text{TP}+\text{FN}} \tag{19}$$

$$\text{FPR}=\frac{\text{FP}}{\text{TN}+\text{FP}} \tag{20}$$

根据文献[3]进行第一次评估，将每个算法的精度与随机预测器的精度进行比较，其中随机预测器的精度为图形时刻 t+1 所创建新链接的数量除以图形时刻 t 消失链接的数量。每个数据集中随机预测器的精度如表 4 所示。该指标也称为 Top-n 精度，其中 n 表示时刻 t 节点之间增加的新链接。在本文实验中，所有算法都进行了 100 次迭代实验，并取平均值作为预测精度的实验结果，且各组方差的分布区间为[1.033, 3.784]。

根据优于随机预测器的倍数，SE-ACO 算法的 Top-n 与其他算法的精度比较情况如表 5 所示，n 为大于或等于 1 的正整数，精度会随 n 先变大再变小。对于每个预测器，给定数字表示优于随机预测器的倍数。例如新浪微博数据集上 SE-ACO 算法的结果为 42.68，这表明该算法的精度要优于随机预测器 42.68 倍，因此 SE-ACO 算法在新浪微博中的精度为 24.69%，即 0.578 4%与 42.68 的乘积。表 5 中的 SR 和 RP 用于大型数据库时非常耗时，故使用少数数据集进行实验时未使用这些算法。对于 Twitter 数据集的实验结果中 AA 预测器效果最佳的原因，本文认为这是由于 AA 预测器为基于节点相似度进行预测的，其中度数较小的 2 个节点的公共邻点比其他节点更有价值。而由于 Twitter 为主要发布短文本的社交网络平台，故节点之间相似度本身较强，预测精度相对较高。

本文实验中，由于使用 SE-ACO 算法预测列表在每次运行时都有可能不同，故设置评估结果为迭代 100 次运行后的平均值，运行结果的标准差为 1.328。SE-ACO 算法考虑了图 1 中的子图(a)与子图(b)。

蚂蚁的初始数量取决于图的节点数，如果蚂蚁数量不足，则所找到三角关系可能较少，从而降低了 SE-ACO 算法的评估结果。实验表明，当蚂蚁的初始数量设定为网络节点数的 10～20 倍时，SE-ACO 算法的运行效率最佳。本文实验将蚂蚁的初始数量设为各数据集节点数的 10 倍。为进一步

表 4 几种算法在数据集的精度对比

算法	新浪微博	Twitter	Facebook	hep-ph	astro-ph	dblp-collab	dblp-cite	polblogs	patent-colla
随机预测器精度	57.84%	69.42%	56.92%	24.81%	49.02%	3.71%	5.89%	37.84%	7.38%
CN	46.73%	57.28%	43.81%	31.90%	48.91%	3.81%	3.94%	39.82%	4.83%
AA	54.28%	64.90%	48.94%	49.39%	39.30%	12.21%	6.44%	48.39%	12.93%
JC	59.39%	68.38%	59.10%	38.18%	42.89%	4.93%	4.92%	43.82%	4.83%
PA	48.30%	54.89%	52.78%	39.28%	42.19%	4.37%	5.84%	43.92%	3.20%
Katz	30.34%	43.99%	58.94%	21.03%	43.90%	5.49%	4.83%	22.07%	0.28%
Distance	56.59%	54.89%	64.90%	58.39%	46.37%	27.83%	4.83%	40.91%	3.35%
RP	68.59%	64.72%	69.28%	43.89%	54.19%	5.44%	5.91%	58.42%	0.43%
SR	55.84%	44.83%	58.49%	56.92%	34.90%	12.83%	6.45%	44.90%	20.34%
PF	34.23%	54.30%	53.91%	41.02%	23.81%	7.38%	5.48%	42.31%	11.23%
RA	53.25%	43.90%	57.84%	37.26%	17.83%	8.65%	9.43%	35.08%	3.34%
SE-ACO	**89.43%**	**91.43%**	**85.49%**	**87.84%**	**89.02%**	**78.44%**	**73.92%**	**86.85%**	**82.57%**

注：加粗的数字表示针对某数据集精度最高的算法。

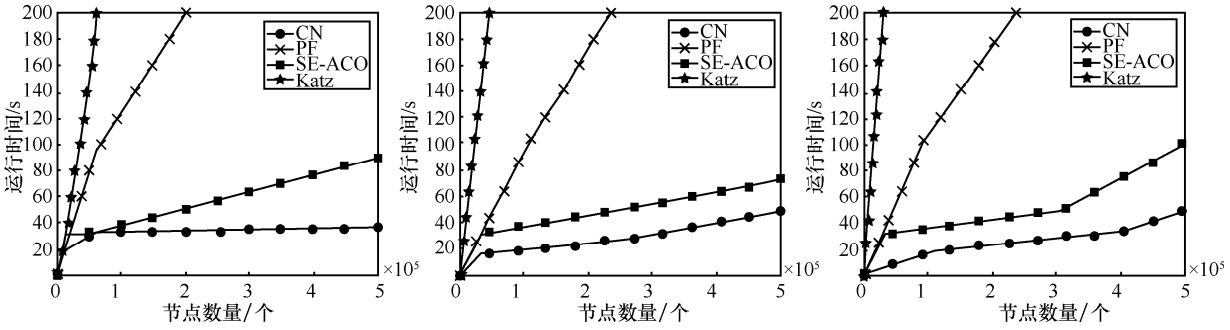
比较算法的可扩展性，本文比较了不同网络中基于节点数量的算法的运行时间，如图 5 所示。由图 5 可知，运行效率最高的算法为 CN 算法，而 SE-ACO 算法的运行时间接近 CN 算法，这表明它可用于大型数据集的预测。且各预测器在不同数据集的表现基本一致，故在此不再赘述。

其他评估方法为 ROC 面积[49]和精确率−召回率曲线方法[50]。各算法根据 ROC 曲线下面积的评估结果如表 6 所示；新浪微博、hep-ph 和 patent-colla 数据集根据精确率−召回率曲线下面积所得的评估结果如表 7 所示。ROC 曲线如图 6(a)～图 6(c)所示，所选的 3 个链路预测算法在 3 个数据集上的精确率−召回率曲线如图 6(d)～图 6(f)所示。结果表明，在大多数预测器中 SE-ACO 算法的精度都高于其他算

表 5　　SE-ACO 算法的 Top-*n* 与其他算法的精度比较情况

算法	新浪微博	Twitter	Facebook	hep-ph	astro-ph	dblp-collab	dblp-cite	polblogs	patent-colla
CN	36.73	168.93	73.91	26.89	17.83	4 693	284.03	68.48	4737
AA	29.89	**193.78**	50.84	18.04	13.29	4 791	419.02	56.03	4290
JC	38.91	122.9	68.05	23.91	19.24	3 980	382.99	64.83	4379
PA	24.73	96.09	49.55	21.64	17.83	3 678	376.91	61.2	4728
Katz	19.04	152.66	56.04	27.9	**26.04**	4 017	392.04	65.09	4910
Distance	10.87	103.94	47.82	24.5	18.75	3 913	386.91	63.48	4289
RP	27.98	84.07	54.17	28.99	23.89	4 692	428.94	**78.93**	—
SR	24.01	79.3	48.91	25.63	17.32	—	217.03	47.16	—
PF	31.07	105.87	68.93	27.57	20.88	4 903	472.04	73.98	5382
RA	34.97	107.41	70.66	29.17	21.94	5 038	480.94	75.26	5490
SE-ACO	**42.68**	162.98	**84.37**	**34.81**	25.46	**5 194**	**529.4**	72.63	**5629**

注：加粗的数字表示针对某数据集精度最高的算法。



（a）新浪微博　　（b）hep-ph　　（c）patent-colla

图 5　算法运行时间对比

表 6　　基于 ROC 曲线下面积的不同算法比较

算法	新浪微博	Twitter	Facebook	hep-ph	astro-ph	dblp-collab	dblp-cite	polblogs	patent-colla
CN	57.83%	51.83%	58.37%	59.02%	52.91%	59.33%	63.89%	67.33%	58.31%
AA	57.31%	58.94%	52.09%	56.38%	58.44%	53.83%	62.91%	63.40%	68.37%
JC	59.75%	57.21%	52.98%	56.12%	52.89%	51.92%	69.23%	63.71%	58.19%
PA	58.38%	50.76%	50.38%	59.83%	64.92%	50.33%	58.92%	59.43%	58.93%
Katz	58.49%	52.93%	51.82%	62.98%	**60.42%**	61.24%	61.20%	70.26%	**69.04%**
Distance	50.00%	50.00%	50.00%	50.00%	50.00%	50.00%	50.00%	50.00%	50.00%
RP	62.83%	57.29%	39.40%	53.72%	58.93%	60.22%	64.95%	67.29%	—
SR	58.03%	**61.89%**	49.38%	60.41%	53.78%	—	68.94%	70.32%	—
PF	53.82%	60.44%	48.39%	53.24%	48.91%	52.31%	57.43%	67.93%	61.22%
RA	61.05%	48.92%	42.94%	54.10%	58.93%	61.84%	60.11%	59.35%	54.89%
SE-ACO	**68.43%**	60.28%	**63.82%**	**64.27%**	60.32%	**63.91%**	**72.81%**	**87.83%**	62.39%

注：加粗的数字表示针对某数据集精度最高的算法。

表 7　　基于精确率–召回率曲线下面积的不同算法比较

算法	新浪微博	**Twitter**	Facebook	hep-ph	astro-ph	dblp-collab	dblp-cite	polblogs	patent-colla
CN	3.28%	**7.81%**	3.66%	0.83%	2.89%	1.62%	5.74%	5.71%	1.74%
AA	4.49%	4.10%	4.07%	1.56%	1.63%	2.18%	4.23%	6.18%	2.38%
JC	2.71%	5.89%	2.90%	0.94%	1.78%	1.37%	3.84%	5.41%	2.91%
PA	1.93%	4.90%	6.85%	1.72%	**4.83%**	2.71%	4.12%	2.37%	3.72%
Katz	2.31%	3.72%	8.57%	2.17%	1.29%	1.63%	5.33%	3.92%	2.81%
Distance	1.52%	1.87%	4.95%	0.73%	0.74%	2.65%	4.81%	4.36%	0.84%
RP	3.93%	4.38%	7.83%	0.65%	1.58%	3.28%	4.73%	4.01%	—
SR	4.25%	3.99%	9.12%	1.22%	2.34%	—	4.10%	**7.41%**	—
PF	2.84%	5.84%	8.49%	0.93%	2.45%	2.81%	4.61%	5.26%	0.73%
RA	0.83%	3.42%	39.28%	0.34%	2.04%	4.09%	5.19%	4.31%	1.27%
SE-ACO	**6.74%**	7.15%	**14.34%**	**2.67%**	4.49%	**4.22%**	**6.22%**	7.34%	**5.64%**

注：加粗的数字表示针对某数据集精度最高的算法。

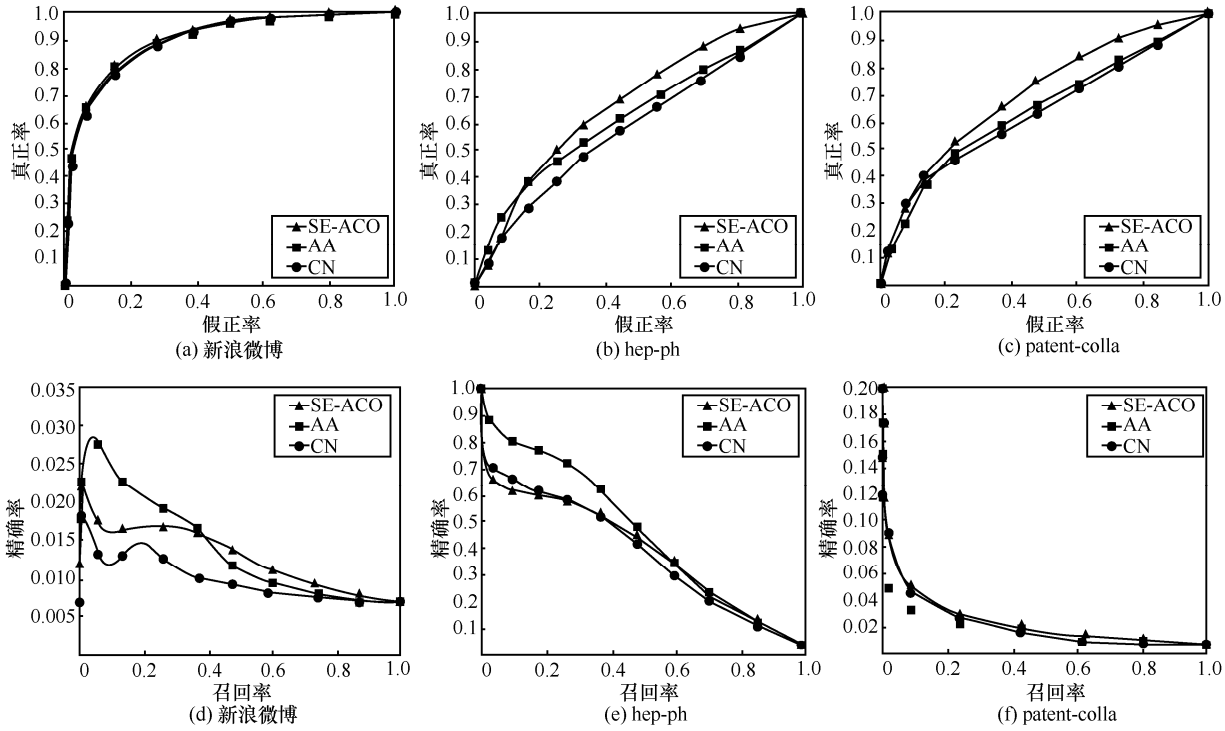


图 6　基于 ROC 与精确率–召回率曲线的算法比较结果

法，而预测器在某几个数据集中精度较低的原因是由于数据集本身特征所导致（其他预测器实验所得结果在此不再赘述）。

表 4 表明可用随机链路预测器的性能评估链路预测算法的总体性能，即如果随机预测器在某些网络上表现良好，则链路预测算法在该网络上的性能也较好。另一方面，随机预测器在聚类系数较高、密度较高且图形直径较短的网络上的性能较好（如表 2 所示）。Katz 指数性能取决于网络直径，即在直径较短的网络中，该算法有着更好的 Top-*n* 精度（如表 2 和表 5 所示）。AA 算法与 CN 算法均使用了公共节点数来测量相似度，但 AA 算法几乎在所有数据集上均优于 CN 算法。AA 算法在聚类系数较高的网络中有着较好的性能，其原因是在这些算法中，能将更多的三元组转化为三角关系的链接为价值较高的链接，其得分也较高（如表 2 和表 6 所示）。SE-ACO 算法在 dblp-collab、dblp-cite 和 patent-collab 数据集上的结果较好，其主要原因为该数据集的 SCC 值较高，因此使用节点度数和路径长度方法的性能较低。在 SE-ACO 算法中，蚂蚁开始时是随机分散在社交网络的各个部分，更多地利用网络的结构特性，因此 SE-ACO 算法在出度较高

的网络上有着较好的性能。图 6 表示 SE-ACO 算法的链路预测精度较高，这是由于根据图 1 中子图(b)所预测的链接有较高的分数，这些链接位于预测列表的顶部，但根据子图(a)所预测的结果则较差。表 6 中的 Distance 算法在所有数据集上所得结果均一致，这是因为该算法将它预测的所有链接都关联了相同的分数。最后本文将表 5 与表 7 中的结果进行对比可知，如果这些算法用 Top-n 精度得出的性能较好，则其用精确度–召回率曲线下面积所得出的性能也较好。

为评估 SE-ACO 算法在大型公开标准数据集中的性能，本文选择了 3 个数据集：MovieLens 1M、MovieLens 10M 基准数据集[51]和 Epinion 数据集[52]，结果如图 7 所示。由图 7 可知，在大型公开标准数据集中，SE-ACO 算法较其他算法仍然可以得到较高的精度，这也再次证明了 SE-ACO 算法的科学性。

5　结束语

本文提出了一种基于子图演化与改进蚁群优化算法的社交网络链路预测方法。首先在社交网络图中确定特殊子图；然后研究子图演化以预测图中的新链接，并用蚁群优化算法定位特殊子图；最后本文针对 SE-ACO 算法使用不同网络拓扑环境与数据集进行检验与算法比较。实验结论表明，与其他无监督社交网络预测算法相比，SE-ACO 算法在多数数据集上的评估结果最好，这表明图形结构在链路预测算法中起到重要作用。SE-ACO 算法在大型公开标准数据集上的运行时间较短且效果较佳。通过使用 SE-ACO 算法，能以高度并行方式进行链路预测。

本文可以从以下几个方面进一步展开研究。1) 由于数据可得性，本文与众多文献[7,44]一样，采用若干个社交网络数据集进行实验，这些数据集往往具有不同的量级，这会在一定程度上影响实验结果[45]。因此，在未来的研究中，有必要使用不同场景、不同量级的数据集来进行社交网络中的链路预测实验。2) 本文仅包含图 1 所示的 2 种子图结构，未来可以尝试使用更复杂的子图结构进行算法改进，这能够使链路预测算法适合更多的社交网络场景。3) 未来可以进一步融入其他机器学习和深度学习方法进行算法改进。

参考文献：

[1] 李永立, 罗鹏, 张书瑞. 基于决策分析的社交网络链路预测方法[J]. 管理科学学报, 2017, 20(1): 64-74.
LI Y L, LUO P, ZHANG S R. Link prediction in social networks based on decision analysis[J]. Journal of Management Sciences in China, 2017, 20(1): 64-74.

[2] WANG Z, LIANG J, LI R. A fusion probability matrix factorization framework for link prediction[J]. Knowledge-Based Systems, 2018, 159: 72-85.

[3] LIBEN-NOWELL D, KLEINBERG J. The link-prediction problem for social networks[J]. Journal of the American Society for Information Science, 2007, 58(7): 1019-1031.

[4] 王智强, 梁吉业, 李茹. 基于信息融合的概率矩阵分解链路预测方法[J]. 计算机研究与发展, 2019, 56(2):306-318.
WANG Z Q, LIANG J Y, LI R. Probability matrix factorization for link prediction based on information fusion[J]. Journal of Computer Research and Development, 2019, 56(2): 306-318.

[5] HUANG Z, LIN D K J. The time-series link prediction problem with applications in communication surveillance[J]. Informs Journal on Computing, 2008, 21(2): 286-303.

[6] YIN D, HONG L, DAVISON B D. Structural link analysis and prediction in microblogs [C]//Proceedings of the 20th ACM International Conference on Information and Knowledge Management. New York: ACM Press, 2011:1163-1168.

[7] PECH R, HAO D, LEE Y L, et al. Link prediction via linear optimization[J]. Physica A: Statistical Mechanics and Its Applications, 2019, 528: 121319.

[8] JACCARD P. Étude comparative de la distribution florale dans une portion des Alpes et des Jura[J]. Bull Soc Vaudoise Sci Nat, 1901, 37: 547-579.

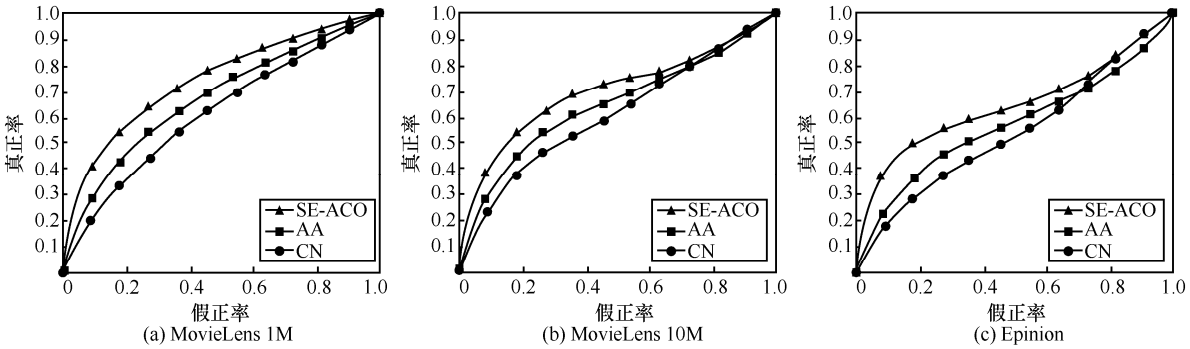


图 7　公开标准数据集中基于 ROC 曲线下面积的几种算法的比较结果

[9] HU H, ZHU C, AI H, et al. LPI-ETSLP: lncRNA–protein interaction prediction using eigenvalue transformation-based semi-supervised link prediction[J]. Molecular Biosystems, 2017, 13(9): 1781-1787.

[10] 王守辉, 于洪涛, 黄瑞阳, 等. 基于模体演化的时序链路预测方法[J]. 自动化学报, 2016, 42(5):735-745.
WANG S H, YU H T, HUANG R Y, et al. Time series link prediction method based on phantom evolution[J]. Acta Automatica Sinica, 2016, 42(5): 735-745.

[11] GAO H, HUANG J B, CHENG Q, et al. Link prediction based on linear dynamical response[J]. Physica A: Statistical Mechanics and Its Applications, 2019, 527: 121397.

[12] WU J H, SHEN J, ZHOU B, et al. General link prediction with influential node identification[J]. Physica A: Statistical Mechanics and its Applications, 2019, 523: 996-1007.

[13] 王凯, 李星, 兰巨龙, 等. 一种基于资源传输路径拓扑有效性的链路预测方法[J]. 电子与信息学报, 2020, 42(3):653-660.
WANG K, LI X, LAN J L, et al. A new link prediction method for complex networks based on topological effectiveness of resource transmission paths[J]. Journal of Electronics & Information Technology, 2020, 42(3): 653-660.

[14] 舒坚, 张学佩, 刘琳岚, 等. 基于深度卷积神经网络的多节点之间链路预测方法[J]. 电子学报, 2018, 46(12): 2970-2977.
SHU J, ZHANG X P, LIU L L, et al. Multi-nodes link prediction method based on deep convolution neural networks[J]. Acta Electronica Sinica, 2018, 46(12): 2970-2977.

[15] DORIGO M, BLUM C. Ant colony optimization theory: a survey[J]. Theoretical Computer Science, 2005, 344(2-3): 243-278.

[16] FADAEE S A, AMIR H M. Classification using link prediction[J]. Neurocomputing, 2019, 359: 395-407.

[17] LICHTENWALTER R N, CHAWLA N V. Vertex collocation profiles: subgraph counting for link analysis and prediction [C]//Proceedings of the 21st International Conference on World Wide Web. New York: ACM Press, 2012:1019-1028.

[18] 张子柯. 在线社交网络信息传播机制与动力学研究综述[J]. 情报学报, 2017, 36(4):422-431.
ZHANG Z K. Mechanisms and dynamics of information spreading on online social networks: a state-of-the-art survey[J]. Journal of the China Society for Scientific and Technical Information, 2017, 36(4): 422-431.

[19] ZHANG Q M, LYU L, WANG W Q, et al. Potential theory for directed networks[J]. PLos One, 2013, 8(2): e55437.

[20] 胡文斌, 王欢, 严丽平, 等. 混合指标量子群智能社会网络事件检测方法[J]. 软件学报, 2016, 27(11):2747-2762.
HU W B, WANG H, YAN L P, et al. Hybrid quantum swarm intelligence indexing for event detection in social networks[J]. Journal of Software, 2016, 27(11): 2747-2762.

[21] 郭丽媛, 王智强, 梁吉业. 基于边重要度的矩阵分解链路预测算法[J]. 模式识别与人工智能, 2018, 31(2):150-157.
GUO L Y, WANG Z Q, LIANG J Y. Link prediction algorithm by matrix factorization based on importance of edges[J]. Pattern Recognition and Artificial Intelligence, 2018, 31(2): 150-157.

[22] GONG N Z, TALWALKAR A, MACKEY L, et al. Joint link prediction and attribute inference using a social-attribute network[J]. ACM Transactions on Intelligent Systems and Technology, 2014, 5(2): Article 27.

[23] GORI M, MONFARDINI G, SCARSELLI F. A new model for learning in graph domains[C]//Proceedings of the 2005 IEEE International Joint Conference on Neural Networks. Piscataway: IEEE Press, 2005: 729-734.

[24] BRONSTEIN M M, BRUNA J, LECUN Y, et al. Geometric deep learning: going beyond euclidean data[J]. IEEE Signal Processing Magazine, 2017, 34(4): 18-42.

[25] 白铂, 刘玉婷, 马驰骋, 等. 图神经网络[J]. 中国科学:数学, 2020, (3): 367-384.
BAI B, LIU Y T, MA C P, et al. Graph neural network[J]. Scientia Sinica (Mathematica), 2020, (3): 367-384.

[26] FAN W Q, MA Y, LI Q, et al. Graph neural networks for social recommendation[C]//The World Wide Web Conference. New York: ACM Press. 2019:417-426.

[27] 郭嘉琰, 李荣华, 张岩, 等. 基于图神经网络的动态网络异常检测算法[J]. 软件学报, 2020, 31(3): 748-762.
GUO J Y, LI R H, ZHANG Y, et al. Graph neural network based anomaly detection in dynamic networks[J]. Journal of Software, 2020, 31(3): 748-762.

[28] 李冬, 申德荣, 寇月, 等. 基于层次化混合特征图的链路预测方法[J]. 中国科学（信息科学）, 2020, 50(2): 221-238.
LI D, SHEN D R, KOU Y, et al. Research on a link-prediction method based on a hierarchical hybrid-feature graph[J]. Science in China(Information Sciences), 2020, 50(2): 221-238.

[29] 方哲, 游宏梁, 薛非, 等. 专家知识协作加权超网络模型及其超链路预测研究[J]. 科研管理, 2017, 38(S1): 259-266.
FANG Z, YOU H L, XUE F, et al. Research on expert knowledge collaboration weighted super network model and hyperlink prediction[J]. Science Research Management, 2017, 38(S1): 259-266.

[30] 尚凤军, 龚文娟, 耿哲. 基于链路预测和网络编码的 MAC 机制[J]. 通信学报, 2016, 37(1): 17-27.
SHANG F F, GONG W J, GENG Z. MAC mechanism based on link prediction and network coding[J]. Journal on Communications, 2016, 37(1): 17-27.

[31] HUANG Z. Link prediction based on graph topology: the predictive value of generalized clustering coefficient[J]. SSRN Electronic Journal, 2010:1634014.

[32] 黄璐, 朱一鹤, 张嶷. 基于加权网络链路预测的新兴技术主题识别研究[J]. 情报学报, 2019, 38(4): 335-341.
HUANG L, ZHU Y H, ZHANG Y. Research on identification of emerging topics based on link prediction with weighted networks[J]. Journal of the China Society for Scientific and Technical Information, 2019, 38(4): 335-341.

[33] LATAPY M. Main-memory triangle computations for very large (sparse (power-law)) graphs[J]. Theoretical Computer Science, 2008, 407(1): 458-473.

[34] TSOURAKAKIS C E, DRINEAS P, MICHELAKIS E, et al. Spectral counting of triangles via element-wise sparsification and triangle-based link recommendation[J]. Social Network Analysis and Mining, 2011, 1(2): 75-81.

[35] 刘树新, 李星, 陈鸿昶, 等. 基于资源传输匹配度的复杂网络链路预测方法[J]. 通信学报, 2020, 41(6): 70-79.
LIU S X, LI X, CHEN H C, et al. Link prediction method based on matching degree of resource transmission for complex network[J]. Journal on Communications, 2020, 41(6): 70-79.

[36] NEWMAN M E J. Clustering and preferential attachment in growing networks[J]. Physical Review E: Statistical Nonlinear and Soft Matter

Physics, 2001, 64(2): 25102.

[37] WU B, WU G, YANG M. A MapReduce based ant colony optimization approach to combinatorial optimization problems[C]//Proceedings of the 2012 8th International Conference on Natural Computation. Piscataway: IEEE Press, 2012: 728-732.

[38] DEAN J, GHEMAWAT S. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008, 51(1): 107-113.

[39] NEWMAN M E J. Assortative mixing in networks[J]. Physical Review Letters, 2002, 89(20): 208701.

[40] LICHTENWALTER R N, CHAWLA N V. Lpmade: link prediction made easy[J]. The Journal of Machine Learning Research, 2011, 12: 2489-2492.

[41] ADAMIC L A, GLANCE N. The political blogosphere and the 2004 U.S. election: divided they blog [C]//Proceedings of the 3rd International Workshop on Link Discovery. Chicago: Association for Computing Machinery, 2005:36-43.

[42] LESKOVEC J, KLEINBERG J, FALOUTSOS C. Graphs over time: densification laws, shrinking diameters and possible explanations[C]// Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining. New York: ACM Press, 2005: 177-187.

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

[44] BRIN S, PAGE L. The anatomy of a large-scale hypertextual Web search engine[J]. Computer Networks, 1998, 30: 107-117.

[45] JEH G, WIDOM J. SimRank: a measure of structural-context similarity [C]// Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2002: 538-543.

[46] LICHTENWALTER R N, LUSSIER J T, CHAWLA N V. New perspectives and methods in link prediction [C]//Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2010: 243-252.

[47] 李勇军, 尹超, 于会, 等. 基于最大熵模型的微博传播网络中的链路预测[J]. 物理学报, 2016, 65(2): 31-41.
LI Y C, YIN C, YU H, et al. Link prediction in microblog retweet network based on maximum entropy model[J]. Acta Physica Sinica, 2016, 65(2): 31-41.

[48] 翟东升, 刘鹤, 张杰, 等. 一种基于链路预测的技术机会挖掘方法[J]. 情报学报, 2016, 35(10):1090-1100.
ZHAI D S, LIU H, ZHANG J, et al. Approach to mining technology opportunity based on link prediction[J]. Journal of the China Society for Scientific and Technical Information, 2016, 35(10): 1090-1100.

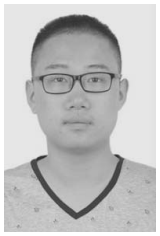
[49] HAND D J. Measuring classifier performance: a coherent alternative to the area under the ROC curve[J]. Machine Learning, 2009, 77(1): 103-123.

[50] DAVIS J, GOADRICH M. The relationship between precision-recall and ROC curves[C]//Proceedings of the 23rd International Conference on Machine Learning. New York: ACM Press, 2006: 233-240.

[51] 肖婧, 张永建, 许小可. 复杂网络模糊重叠社区检测研究进展[J]. 复杂系统与复杂性科学, 2017, 14(3): 8-29.
XIAO J, ZHANG Y J, XU X K. Research progress of fuzzy overlapping community detection in complex networks[J]. Complex Systems and Complexity Science, 2017, 14(3): 8-29.

[52] 叶小莺, 万梅, 唐蓉, 等. 基于图聚类与蚁群优化算法的社交网络聚类算法[J]. 计算机应用研究, 2020, 37(6): 1670-1674, 1687.
YE X Y, WAN M, TANG R, et al. Clustering algorithm of social network based on graph clustering and ant colony optimization algorithm[J]. Application Research of Computers, 2020, 37(6): 1670-1674,1687.

[作者简介]



顾秋阳（1995− ），男，浙江杭州人，浙江工业大学博士生，主要研究方向为智能信息处理、数据挖掘、中小企业高质量发展等。

琚春华（1962− ），男，博士，浙江衢州人，浙江工商大学教授、博士生导师，主要研究方向为智能信息处理、数据挖掘、电子商务与物流优化等。

吴功兴（1974− ），男，博士，浙江义乌人，浙江工商大学副教授，主要研究方向为智能信息处理、数据挖掘、电子商务与物流优化等。