

基于粒子群算法的 Web 社区发现^{*}

段晓东^{1,2} 王存睿² 刘向东² 林延平³

(东北大学信息科学与工程学院 沈阳 110004)¹ (大连民族学院非线性信息技术研究所 大连 116600)²
(加拿大 Alberta 大学数学科学系 埃德蒙顿 T6G 2G1)³

摘要 Web 社区发现技术是提高网络搜索引擎检索质量的重要途径之一。如何给出利用较少先验信息,并能对网络进行高效划分的算法是网络社区发现的关键。传统算法如 Wu-Huberman 算法虽能对社区进行快速划分,但需先确定分属不同社区的两个节点, Radichi 快速分裂算法依赖于网络中存在的三角形的数目, Duch J 提出的极值优化算法对初始解非常敏感。本文提出一种基于粒子群优化算法的网络社区发现方法,并用不同规模的网络图 Zachary、Krebs 和 dolphins 网络结构对方法进行测试,实验结果表明,该方法在无先验信息的条件下,以较低的时间复杂度,快速、高效地完成对网络社区的划分。

关键词 Web 社区, 社区发现, 粒子群算法, 搜索引擎

Web Community Detection Model Using Particle Swarm Optimization

DUAN Xiao-Dong^{1,2} WANG Cun-Rui² LIU Xiang-Dong² LIN Yan-Ping³

(Faculty of Information Science and Engineering of Northeastern University, Shenyang 110004)¹

(The Research Institute of Nonlinear Information Technology of Dalian Nationalities University, Dalian 116600)²

(Department of Mathematical & Statistical Science of Alberta University, Edmonton, T6G 2G1)³

Abstract Web community detection is one of important ways to enhance retrieval quality of Web search engine. How to design one highly effective algorithm to partition network community with few domain knowledge is first the key to network community detection. Traditional algorithms, such as Wu-Huberman algorithm can carry on the fast partition to the community, but must have known two nodes belonging to the different communities. In addition, the Radichi algorithm relies on the triangle number which exists in the network, the extremal optimization algorithm proposes by Duch J. is extremely sensitive to the initial solution, easy to fall into the local optimum. This article proposes a new model based on particle swarm optimization to detect network community, and with different scale network chart, such as Zachary, Krebs and dolphins network architecture to test the algorithm, the experimental results indicate this model can fast, effectively search global optimum partition of network structure without any domain information.

Keywords Web community, Community detection, Particle swarm optimization, Search engine

1 引言

目前,互联网正朝着社区化的方向发展,用户希望通过参与、互动,获得个性化、可信任的信息。由于社区中网站的内容所涵盖的领域专业性和关联性较强,因此如何识别、分析和利用网络社区信息成为下一代搜索引擎技术发展的关键。从 Web 网站通过链接形成的网络和其中用户所形成的关系网中划分出不同的社区,主要依靠社区划分算法进行解决。社区划分问题和图分割问题(Graph Partition)有着密切的联系,一般情况下,这类社区划分问题的精确求解为一个 NP 难题^[1]。因此,此方面的工作一直受到国内外研究的关注^[2~5]。

2 Web 网络社区结构发现算法

社区发现方法比较著名的包括 Kernighan 算法和基于 Laplace 图特征值的谱平分法。Kernighan 算法是一种基于贪

婪算法原理^[6],将网络分割为两个大小已知的社区的二分法,但该算法必须已知网络社区的确切规模才能得到正确结果,使其在实际网络分析中难以得到较好的应用。基于 Laplace 图特征值的谱平分法,其利用网络结构的 Laplace 矩阵中不为零的特征值所对应的特征向量和同一个社区内的结点对应的元素近似相等的原理对网络社区进行划分,在规模为 n 节点的网络中,该算法的算法复杂度等同于求 $n \times n$ 矩阵的特征向量的复杂度,为 $O(n^3)$ 。Wu 和 Huberman 提出了基于电阻网络电压谱的快速分割算法^[4],它将网络看成一个电阻网络,然后,利用 Kirchhoff 定理求解各个节点的电压值,绘制电压谱,进而根据不同的阈值进行社区划分。该算法的复杂度只有 $O(n+m)$,其中 n 表示网络节点数目, m 为网络中边的数量,如图 1(a)所示,但该方法需已知分属于不同社区的两个节点。

除上述算法之外,还有利用分级聚类概念提出的分裂算法和凝聚算法。其中的 GN 算法^[7]通过不断地从网络中移除

^{*} 国家自然科学基金(No. 60573124);教育部优秀青年教师资助计划;辽宁省自然科学基金资助。段晓东 博士,教授,主要研究方向为人工智能与非线性信息处理技术;王存睿 讲师,主要研究方向为智能优化算法和计算复杂性;刘向东 博士,教授,主要研究方向为非线性信息处理和算法复杂性;林延平 博士生导师,教授。

介数(网络中经过每条边的最短路径的数目)最大的边,逐步将整个网络分成越来越小的各部分。GN算法的不足在于对网络社区划分优劣没有一个定量描述。因此,Newman等人经过研究提出了一种度量网络社区划分质量的标准,成功地解决了这个问题^[8]。Radicchi等人在GN算法基础上提出快速分裂算法^[9],通过计算局部变量,减小了全局的运算量,但该算法本身很大程度依赖于网络中存在的三角形的数量。

Web网络如果存在的三角形的数目很少(如图1(b)中所示的网络,其中含三角形结构数目较少),那么相应边的聚类系数就会很小,此算法就无法正确地搜索网络中的社区结构。J. Duch提出了一种社区划分的极值优化算法^[10],其通过调整局部极值来优化全局最优的变量,进而提高算法效率。但此算法的搜索结果很大程度依赖于网络初始的划分情况,还会出现如图1(c)的孤立点或不连通的局部最优的社区结构。

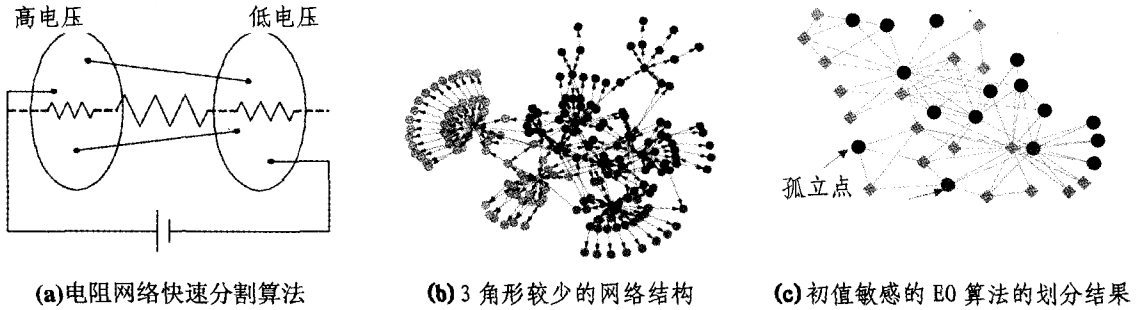


图1 传统网路社区划分算法图示

通过对传统网络社区发现算法的分析可以看出,设计一种无需先验信息且问题不相关、时间复杂度较低的算法,进而对各种规模的网络进行划分是十分必要的,且具有较现实的应用价值。

3 基于粒子群算法的网络社区划分方法

粒子群算法作为一种基于群智能的高效优化算法,对解决多目标、高维度的复杂优化问题已显示出一定的优势^[11]。本文提出的基于粒子群算法的网络社区发现方法需首先解决以下三个问题:一是定义衡量网络社区划分优劣的度量函数,作为粒子的适值函数;二是改进粒子群算法解决社区划分这种离散优化问题;三是修复在搜索过程中产生的社区孤立点,确保搜索到合理的解。

3.1 社区划分的粒子群算法

粒子群优化算法是由Kennedy和Eberhart在1995年提出的一种基于群智能的演化计算技术^[11],是在鸟群、鱼群和人类社会的行为规律启发下提出的,已被证明是一种非常快速、高效的智能优化算法^[12]。假设在 D 维目标搜索空间中, n 个粒子组成一个群落,每个粒子 i 包含一个 D 维的位置向量 $x_i=(x_{i1}, x_{i2}, \dots, x_{iD})$ 和速度向量 $v_i=(v_{i1}, v_{i2}, \dots, v_{iD})$ 。粒子 i 在搜索解空间时,记住其搜索到的最优位置 p_i 。在每次迭代中,粒子 i 根据当前速度、自身经验 $p_i=(p_{i1}, p_{i2}, \dots, p_{iD})$ 和群体最优经验 $p_g=(p_{g1}, p_{g2}, \dots, p_{gD})$,调整自己目前的速度向量,进而通过速度调整改变自身的当前位置 x_i 。反复迭代对问题空间进行高效搜索,直到满足停止条件为止。

基本粒子群算法采用如下公式更新粒子状态^[10]:

$$v_{id} = \omega v_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (1)$$

$$x_{id} = x_{id} + v_{id}, i=1, 2, \dots, n, d=1, 2, \dots, D \quad (2)$$

其中,学习因子 c_1 和 c_2 是非负常数; r_1 和 r_2 是取值介于(0, 1)之间的随机数, $v_{id} \in [-v_{max}, v_{max}]$; v_{max} 是常数,根据问题而设定。迭代终止条件根据具体问题一般选为最大迭代次数或(和)粒子群迄今为止搜索到的最优位置满足预定最小适应阈值。其中 $\omega \geq 0$,称为惯性因子^[13]。

基于粒子群算法的Web网络社区划分方法中,每个粒子的位置向量对应一种该网络结构下的划分方案。该方法采用社区分裂策略,首先用该方法将网络社区划分为2个社区,然

后移除两个社区相连的边,对每个子社区进行划分,直到不能划分为止。定义粒子 i 的位置向量 $x_i=(x_{i1}, x_{i2}, \dots, x_{iD})$ $x_{id} \in [-x_{max}, x_{max}]$, x_{id} 表示粒子 i 对应划分方案中的节点 d 的分组情况。速度限定在 $[-v_{max}, v_{max}]$ 。这里采用向量 y_i 来存储粒子 i 的位置向量转化结果,转换函数 $y_{id}=f(x_{id})$ 采用如公式(3)定义的双极型阈值函数,即当 $y_{id}=1$ 表示粒子 i 中第 d 个节点属于社区一的节点, $y_{id}=-1$ 表示其为社区二的节点。

$$y_{id} = \begin{cases} 1 & x_{id} \in [0, x_{max}] \\ -1 & x_{id} \in [-x_{max}, 0] \end{cases} \quad (3)$$

基于粒子群的网络社区划分算法流程如下:

Step 1:设置粒子位置和速度的范围 x_{max} 和 v_{max} ,以及粒子群惯性因子 ω ;根据网络节点的数量设置粒子的位置向量和速度向量的维度,初始化粒子数为 n 的粒子群;

Step 2:复制粒子的当前的位置向量到经验位置,并根据公式(4)每个粒子当前位置的适应度,复制到经验适应度;

Step 3:选出适应度最高的粒子,并将其经验位置向量和经验适应度作分别为群最优位置和群最优经验;

Step 4:根据群体最优位置,每个粒子更新自身的位置向量和速度向量;

Step 5:修复每个粒子所代表网络社区划分中的孤立点(详见3.3节),采用绝对孤立点修复策略或柔性修复策略,相应地改变该图节点的分组,然后将粒子 i 中被修复的节点对应的 x_{id} 置于修复后的分组所在的值区间,并将该节点 d 对应的粒子第 d 维速度 v_{id} 置为0);

Step 6:计算每个粒子的适应度,并与其经验适应度相比较,如果优于其经验,则更新该粒子的经验位置及其适应度;

Step 7:计算出群体中最优粒子,与当前群最优适应度相比较,如果优于当前群最优粒子,则更新群最优位置和群最优适应度;

Step 8:如果满足停止条件则,输出群最优划分方案和适应度,否则,转到Step 4。

3.2 粒子适应度定义

对于社区划分问题,每个粒子代表一种社区划分方案。本文度量粒子位置向量 x_i 转化后的向量 y_i 的适值采用Newman等人提出的一种度量网络社区划分质量的标准^[8]。

对于某种划分形式,假设其将网络划分为 k 个社区,定义一个 $k \times k$ 的对称矩阵 $E = (e_{ij})$,其中元素 e_{ij} 表示网络中连接 i 社区和 j 社区的边的数量占所有边的比例。这里所说的边是指在原始网络中的,是利用完整的原始网络计算的,而非在计算过程中被算法破坏的网络。设矩阵中对角线上各元素之和为 $Trace(E) = \sum_i e_{ii}$,它给出了网络中连接某一个社区内部各节点的边在所有边的数目中所占比例。定义每行中各元素之和为 $a_i = \sum_j e_{ij}$,它表示与第 i 个社区中的节点相连的边在所有边中所占的比例。在此基础上,用公式(4)来定义网络社区划分的衡量标准

$$Q(y) = \sum_i (e_{ii} - a_i^2) = Trace(E) - \|E\|^2 \quad (4)$$

$\|E\|^2$ 表示矩阵 e^2 中所有元素之和。(4)式表明网络中连接两个同种类型的节点的边的比例,减去在同样的社区结构下任意连接这两个节点的边的比例的期望值。 Q 越接近 1,表示社区结构越明显。通过这个度量标准就可以建立起社区划分质量的全局度量函数。

3.3 孤立点修复策略

针对网络社区划分问题,作为一种随机优化算法的粒子群算法在优化过程中,其搜索到的划分方案中有孤立点存在的现象。所谓孤立点就是网络图中其邻接的图节点全部或绝大多数是与其相异组的成员节点。孤立点的存在会影响算法搜索的效率,制定不同强度的修复策略可以提高算法的收敛速度。同时孤立点修复策略对于优化算法而言也是一种抖动策略,它可以使算法避免陷入局部最优。本文基于此提出两种修复策略。

① 绝对孤立点修复策略

如图 2(a)所示,如果网络图中存在与其邻接的所有节点

的分组与其相异的节点,这里称其为绝对孤立点。如在算法搜索过程中,搜索的网络社区划分方案中存在绝对孤立点,则强制其改变分组,转为与其邻接点相同的分组。

② 柔性修复策略

如图 2(b)所示,如果网络图中存在其邻接的绝大多数图节点为与其相异的分组节点,则称其为相对孤立点。如在算法搜索过程中,搜索的划分方案中存在相对孤立点,则找出与该点邻接且同组的节点数目 a ,与其相邻接的异组节点数目为 b ,则定义 $L = b/a$ 。如果 L 大于给定阈值 θ ,则强制其改变分组,否则保持分组不变。

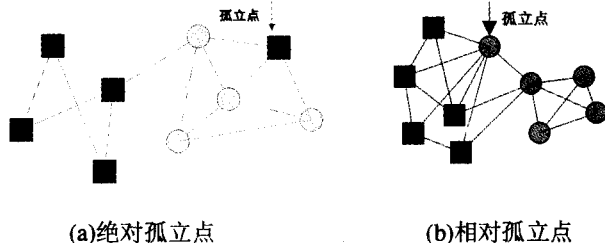


图 2 网络社区中的孤立点

4 测试及结果分析

4.1 图数据集

本文采用 3 个不同规模的网络结构数据对算法进行划分:图 3(a)所示的网络结构是 Wayne Zachary 整理的网络结构,包含 34 个节点,77 条边;图 3(b)所示的为 Lusseau 采集的网络结构^[15],包含 62 个节点,105 条边;图 3(c)为 Krebs 采集的网络结构^[16],包括 152 个节点,449 条边。

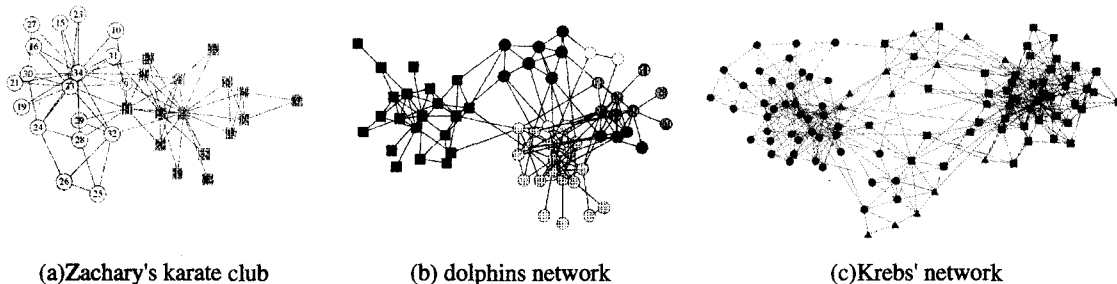


图 3 网络社区算法测试图集

4.2 算法性能测试

4.2.1 收敛速度

本文采用基于修复策略的粒子群网络社区发现方法对 3 个网络结构进行社区划分。种群数目设为 20 个,惯性因子 ω

$= 0.7$,修复策略采用绝对孤立点修复策略或不同阈值 θ 的柔性修复策略,结果如表 1 所示。表 1 列出不同数据集的规模和相应基准划分度量 Q 值,通过 100 次实验,记录下算法的达到基准 Q 的平均迭代次数和平均搜索时间。

表 1 基于粒子群算法的网络社区发现方法的性能

	Karate(34 个节点,77 条边)		dolphins(62 个节点,152 条边)		Krebs(105 个节点,449 条边)	
	达优基准 $Q=0.2317$		达优基准 $Q=0.3315$		达优基准 $Q=0.4127$	
	平均迭代次数	平均搜索时间	平均迭代次数	平均搜索时间	平均迭代次数	平均搜索时间
绝对点修复策略	71.07	329.55 毫秒	271.26	3253.20 毫秒	1906.01	69900.06 毫秒
柔性修复策略						
$\theta=3.5$	41.79	136.72 毫秒	43.26	361.88 毫秒	209.97	4654.71 毫秒
$\theta=3.0$	21.7	77.18 毫秒	38.44	329.08 毫秒	55.06	1394.54 毫秒
$\theta=2.5$	14.45	58.13 毫秒	26.64	234.53 毫秒	30.99	1103.59 毫秒
$\theta=2.0$	7.88	41.71 毫秒	11.52	122.82 毫秒	23.11	1081.45 毫秒
$\theta=1.5$	12.93*	58.43 毫秒*	4.40	78.28 毫秒	11.00	748.29 毫秒
$\theta=1.0$	14.97*	62.34 毫秒*	3.89	77.2 毫秒	7.57	550.93 毫秒

在绝对孤立点修复策略下,从表 1 中可以看出,算法能够完成对网络社区的划分,搜索较优的划分方案,但收敛速度较慢。此方法中,当种群最优值在一定迭代次数内不再增加时就停止迭代。但事实上,搜索结果是否会继续增加是难以定量判断的。以上方法中,如果在 αn 步(n 为网络中的节点数)内没有继续增加,则认为该方法已经到了一个局部最大值,这里 α 是一设定常数。这样,我们可以估计整个算法的时间复杂度。在孤立点强修复策略下,该算法在图节点为 n 、边数为 m 的规模下,时间复杂度为 $O((m+n)\log(n))$ 。

柔性修复策略基于图中某一点的分组与其邻接的节点的分组情况相关这样一种假设。当然不同的网络结构具有不同的相关度。本文针对这 3 个网络结构,分别采用 $\theta=1\sim 3.5$ 不同的取值作为算法的修复策略,并列出了不同策略下的达优的平均迭代时间和平均迭代次数。从表 1 可以看出算法的

平均迭代次数和平均迭代时间在 θ 取 3.5 至 1.5 时,均在减少,这说明柔性修复策略比较明显地提高了算法收敛速度,算法时间复杂度由原来的 $O((m+n)\log(n))$ 降低为 $O((m+n)\log^2(m+n))$,时间复杂度得到很大程度的降低。但分析表 1 中加 * 注的实验数据,算法收敛速度不是随 L 递减而递减,修复强度过大反而影响算法的收敛速度。因此,选取适合强度的孤立点修复策略可以提高算法的效能和收敛速度。

4.2.2 惯性因子 ω 的影响

惯性因子 ω 是粒子群算法的重要调节参数,对算法的收敛结果影响也较大。为观察惯性因子 ω 在网络社区划分问题中的作用,本文经过实验采集了 3 个数据集在不同惯性因子作用下的粒子群适应度变化, ω 由 0.6~0.8,对应不同的取值变化曲线,如图 4 所示。

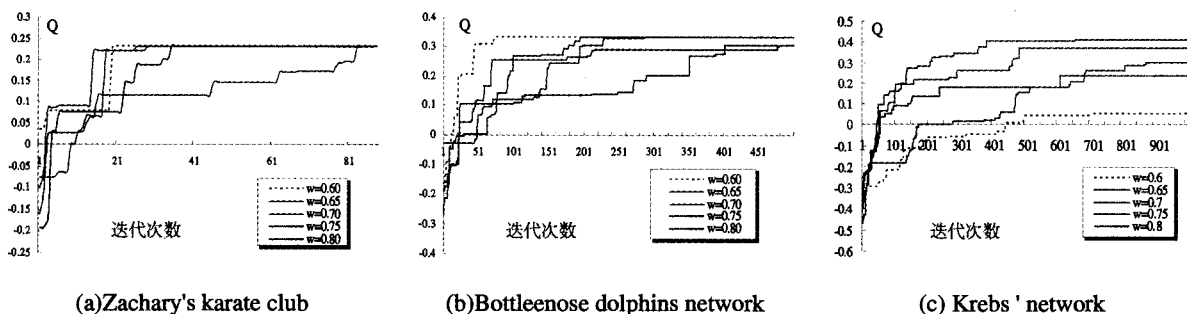


图 4 不同惯性因子 ω 下的适值变化规律

由图 4 中的变化可以看出,不同规模 and 不同结构的数据集需要选定合适的惯性因子。图 4(a)中,可以看出当 $\omega=0.65$ 时,算法收敛较快;在图 4(b)的 dolphins 网络中, $\omega=0.70$ 算法收敛较快;在图 4(c)的 Krebs 结构的适值变化曲线中, $\omega=0.65$ 算法收敛较快。从结果可以分析出,对于不同问题,

选取合适的 ω 可以加快算法收敛速度和精度。

4.2.3 粒子群种群大小的影响

为检验粒子群的粒子规模是否对算法有影响,本文采集了不同种群大小, $swarmsize=10\sim 60$ 的不同图集的适应度变化数据。

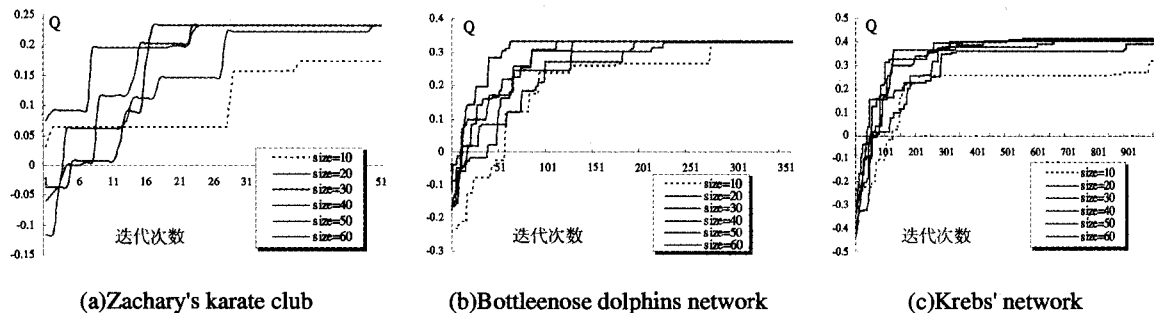


图 5 网络社区算法测试图集

从图 5(a)中可以看出,对于 Zachary 数据集,种群大小为 20~40 时,算法收敛较快,而过大的种群规模并不能提高算法的效率。在图 5(b)中,对于 dolphins 网络,种群大小为 40 时,算法收敛速度较快;当种群大小为 60 时,并没有提高算法的效率。在图 5(c)中,对于 Krebs 网络,种群大小为 20~30 时,算法收敛速度较快。因此,单纯靠提高种群规模并不能相应地提高算法效率。针对网络社区划分问题,种群规模在 20~40,算法就可以得到相对稳定的性能。

实验结果表明,种群规模选择 20~40 比较适宜,过大的种群规模并不能提高算法效率,反而提高了算法的空间复杂度。算法对惯性因子的变化也比较敏感,试验结果表明,惯性因子置于 $[0.65, 0.75]$ 之间,算法可以获得相对稳定的搜索性能。

时间复杂度与其它传统网络社区划分算法相比较,本文提出的柔性修复策略的粒子群网络社区划分方法的算法时间复杂度为 $O((m+n)\log^2(m+n))$,低于 Laplace 图特征值的谱平分法的 $O(n^3)$ 、J. Duch 极值优化算法的 $O(n^2\ln(n))$ 、Radicchi 快速分裂算法的 $O(m^2)$ 。

总结与展望 本文提出了一种基于不同强度修复策略的且时间复杂度为 $O((m+n)\log^2(m+n))$ 的粒子群网络社区发现方法,较好地完成了对网络社区的划分,同时讨论了不同

4.3 结果分析

从以上实验结果可以看出,基于粒子群算法的网络社区发现方法可以有效地对网络社区进行划分,而且可以通过调节孤立点修复强度提高该方法的收敛速度。同时,本文也讨论了不同惯性因子和不同种群规模对社区划分问题的影响。

(下转第 145 页)

由图 3 可知,三种虚拟样本生成方法的精度总体高于原 SVM 分类,其中 Onto-SVM 相对最好,这得益于 Onto-SVM 对样本数量和质量提升的良好表现。但在稳定性方面 Onto-SVM 表现一般,如对 software, movies 的分类 Onto-SVM 在精度上的优势并不突出。分析认为 Onto-SVM 对分类精度的提升作用与本体的领域知识描述质量有很大关系。由于 DOMZ 在 software, movies 的分类颗粒较粗,导致相应构建的本体在领域知识的表达详细程度上不十分理想,因而构造出来的虚拟样本知识蕴含量也相对较低。

3.3.2 推广能力比较

考虑到训练样本数量应当对分类精度有所影响,设计实验依次增加训练样本的数量进行分类精度测试。分别取 50、100、200、400、600、800、1000 个正例(负例数量不变)分别构成训练集,测试四种方法对四类文本的平均 F_1 精度,实验结果如图 4 所示。

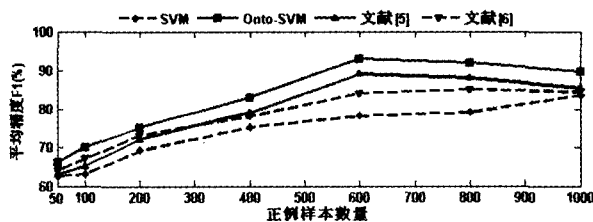


图 4 样本数量与精度比较

由图 4 可知,在不同正例样本数量条件下,虚拟样本构造方法对推广能力的提升作用呈先升后降的趋势,但总体上还是高于原 SVM。其中 Onto-SVM 的增幅最大,而降幅相对最小。分析认为:虚拟样本的构造对原样本数量有很大依赖性,原样本数量越多则生成的虚拟样本的质量也就越好。此外,由于虚拟样本构造具有一定的风险性,很可能构造出与实际不符的样本,我们称之为假样本,会给分类带来危害。随着真样本的增加,假样本的危害作用表现越来越明显。因而可以得出结论:构造虚拟样本的方法更适合小样本空间时的机器

学习,这与文[5]的结论也是一致的。Onto-SVM 的增/降幅的不同表现间接说明其有效样本相对较多。

结束语 本文研究了本体驱动的文本虚拟样本构造,与同类方法相比不同之处在于:利用了本体表示的共享概念化知识作为构造虚拟样本的依据,实验表明该方法是有有效的。未来的工作可以从以下几个方面展开:首先进一步对文中几种虚拟样本的构造方法进行测试,并逐步完善;其次对设计虚拟样本的选择方法进行合理的筛选。随着语义网技术的不断发展,相信本体将会成为越来越重要的知识源驱动,对机器学习进行优化。

参考文献

- 1 苏金树,张博峰,徐昕. 基于机器学习的文本分类技术研究进展[J]. 软件学报,2006,17(9): 1848~1859
- 2 Niyogi P, Girosi F, Poggio T. Incorporating prior information on machine learning by creating virtual examples [J]. Proc. IEEE, 1998, 86(11): 2196~2209
- 3 Poggio T, Vetter T. Recognition and structure from one 2D model view: observations on prototypes, object classes, and symmetries[C]. A. I. Memo No. 1347, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 1992
- 4 Scholkopf B, Simard P, Smola A, et al. Prior knowledge in support vector kernels[C]. Advances in Neural Information Processing Systems. MIT Press, 1998
- 5 李辉,等. 运用文本领域的常识改善基于支撑向量机的文本分类器性能[J]. 中文信息学报,2002, 16(2): 7~13
- 6 Sassano M. Virtual examples for text classification with support vector machines[C]. In: Proceedings of 2003 Conference on Empirical Methods in Natural Language Processing, 2003. 208~215
- 7 Bhogal J, Macfarlane A, Smith P. A review of ontology based query expansion [J]. Information Processing and Management, 2007, 43(4): 866~886
- 8 Latifur R K, McLeod D. Ontology-based information selection [D]. California: University of Southern California, 2000
- 9 Chih-Chung Chang and Chih-Jen Lin. LIBSVM: a library for support vector machines[CP], 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

(上接第 21 页)

强度的修复策略、惯性因子和种群规模对算法的影响,总结了参数和算法性能之间的相互关系。本文提出的基于粒子群算法的网络社区发现方法为一个网络社区二分方法,需要对网络图不断进行分割,完成最后的划分。进一步研究的研究工作可包括 k -社区划分方法等。同时针对在适应度计算随问题规模增长的同时时间复杂度和空间复杂度都跟随增长的问题,可以研究相应的并行粒子群网络社区划分方法、负载均衡该方法的适值计算。这方面的工作有待进行深入的研究。

参考文献

- 1 Garey M R, Johnson D S. Computers and Intractability, A Guide to the Theory of NP-Completeness. San Francisco: W H Freeman, 1979
- 2 王晓宇,周微英. 万维网的链接结构分析及其应用综述[J]. 软件学报,2003,14(10): 1768~1780
- 3 杨楠,弓丹志,李欣,孟小峰. Web 社区发现技术综述[J]. 计算机研究与发展,2005,42(3): 439~447
- 4 Wu F, Huberman B A. Finding communities in linear time: A physics approach[J]. Euro Phys J B, 2003, 38: 331~338
- 5 Radicchi F, Castellano C, Cecconi F, et al. Defining and identifying communities in networks. cond-mat/0309488. 2004
- 6 Kernighan B W, Lin S. An efficient heuristic procedure for partitioning graphs [J]. Bell System Technical Journal, 1970, 49: 291

~307

- 7 Girvan M, Newman M E J. Community structure in social and biological networks [J]. In: Proc Natl Acad, 2002, 99: 7821 ~7826
- 8 Newman M E J, Girvan M. Finding and evaluating community structure in networks [J]. Physical Review E, 2004, 69: 026113
- 9 Radicchi F, Castellano C, Cecconi F, et al. Defining and identifying communities in networks. cond-mat/0309488. 2004
- 10 Duch J, Arenas A. Community detection in complex networks using extremal optimization. Phys Rev E 72, 2005: 027104
- 11 Kennedy J, Eberhart R C. Particle swarm optimization [C]. In: Proceedings of the 1995 IEEE International Conference on Neural Networks, vol 4, IEEE Press, 1942~1948
- 12 Eberhart R C, Shi Y. Comparison Between Genetic Algorithm and Particle Swarm Optimization [C]. Evolutionary Programming VII (1998), Lecture Notes in Computer Science 1447, Springer, 611~616
- 13 Shi Y H, Eberhart R C. A Modified Particle Swarm Optimizer [C]. In: IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, May, 1998
- 14 Zachary W W. An information flow model for conflict and fission in small groups [J]. Journal of Anthropological Research, 1977, 33: 452~473
- 15 Lusseau D. The emergent properties of a dolphin social network [J]. Biology Letters. In: Proc R Soc, London B(suppl.). DOI 10.1098/rsbl.2003.0057. 2003
- 16 Newman M E J. Modularity and community structure in networks [J]. Proc Natl Acad Sci USA 2006(in press)