

改进蚁群算法在文本聚类中的应用研究

吕 岩

(营口职业技术学院, 辽宁 营口 115000)

摘 要: 提出了一种改进蚁群文本聚类算法. 改进蚁群文本聚类算法利用信息素对蚂蚁随机移动进行控制, 使蚂蚁朝着文本向量相对集中的区域移动, 缩短蚂蚁寻找文本向量簇的时间, 提高聚类效率. 采用复旦大学中文文本分类语料库进行仿真实验, 实验结果表明, 改进蚁群文本聚类算法不仅加快了文本聚类算法的收敛速度, 而且提高文本聚类结果的精度.

关键词: 蚁群算法; 文本聚类; 向量空间模型; 信息素

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2012)03-0031-04

Research of Text Clustering Based on Improved Ant Colony Algorithm

LV Yan

(Yingkou Vocational & Technical College, Yingkou 115000, China)

Abstract: This paper proposes a text clustering algorithm based on ant colony algorithm, the ant mobile is controlled by information to text vector of the relative concentration of the area, shorten cluster time, improve the clustering efficiency. The simulation experiment is carried out on the Fudan university Chinese text data, the results show that the proposed text clustering algorithm speeds up the text clustering algorithm convergence speed and improves the accuracy of text clustering results.

Key words: ant colony algorithm; text clustering; vector space model; pheromone

1 引言

在互联网上, 大部分信息以文本形式出现, 文本聚类^[1]是一种无指导的分类方法, 它可以将一个文本集分成若干称为簇的子集, 簇中的成员间相似性比较大, 簇间文本相似性比较小. 当前文本聚类算法有平面聚类法和层次聚类法两大类^[2]. 平面聚类法主要代表算法为: K-means 算法, K-means 算法需要预先设定簇个数 K 值, 在对一个文本进行聚类时, 往往难以预先选择最适合的 K 值, 往往得不到全局最优解, 导致文本聚类精度较低, 给 K-means 算法适用性带来了局限^[3]. 层次聚类法把文本类别看作是有层次的, 形成一棵类别树, 随着数据量的增大, 需要空间比较大, 用忽略数据密度和类别距离大小的不均, 难以获得高精度聚类结果^[4]. 针对文本聚类过程中层出不穷的新问题, 有学者提出一

些新的聚类算法, 如遗传算法, 蚁群算法^[5]. 遗传算法收敛速度慢, 文本聚类效率低; 蚁群算法的蚂蚁行动缺乏目的性, 随机性太大, 影响文本聚类收敛速度和本聚类结果的精确率. 这些不足不仅限制了蚁群聚类算法的发展, 也限制了文本聚类的发展^[6].

针对传统蚁群算法在文本聚类过程存在的不足, 本文提出一种改进蚁群的文本聚类算法, 通过相应文本分类语料库对本文算法的聚类精度进行验证.

2 文本聚类的相关技术

2.1 文本表示

文本是一种非结构化数据, 使用人类的自然语言, 因此在文本聚类之前, 首先应该将文本转换成计算机能够识别的格式, 然后对文本间的相似进行判断, 最后通过聚类算法将文本划分成不同的类别^[7].

本文采用空间向量模型 (Vector Space Model, VSM) 来表示文本. 在 VSM 中, 文本被看作是一组正交特征向量组成的向量空间, 文本中特征项采用一维向量表示, 特征项重要程度通过权重表示, 那么每篇文本 D 都可以映射为特征向量, 即:

$$D = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{bmatrix} = (t_1, t_2, \dots, t_n) \quad (1)$$

$$= \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ w_{m1} & w_{m2} & \dots & w_{mn} \end{bmatrix}$$

式中, d_m 表示第 m 个文本行向量, t_n 表示第 n 个特征项列向量, w_{nm} 表示第 n 个特征值在第 m 个文本中的权重.

在 VSM 中, 文本特征权重一般采用特征词的词频和文本频数来表示, 常采用 $tf \times idf$ 方法来计算, 具体计算公式如式(2)所示.

$$w(t_i, d) = tf(t_i, d) \times \log\left(\frac{N}{df(t_i)} + 0.01\right) \quad (2)$$

由于各文本长度不一样, 这样会对权重的不利影响, 必须对权重值规范到 $[0, 1]$, 最后文本特征权重的计算公式如式(3)所示.

$$w(t_i, d) = \frac{tf(t_i, d) \times \log\left(\frac{N}{N_k} + 0.01\right)}{\sqrt{\sum_{i=1}^n tf^2(t_i, d) \times \log^2\left(\frac{N}{N_k} + 0.01\right)}} \quad (3)$$

2.2 文本相似性度量

为了对一个文本集中的文本聚类, 必须对文本集中文本之间的亲疏远近程度进行描述, 即文本间相似性度量. 在 VSM 型中, 文本相似性的度量主要有相似度函数和距离统计函数两类方法, 考虑到本文使用蚁群算法进行文本聚类分析, 根据蚁群算法的特点, 本文的文本间非相似性度量选用余弦系数来进行, 具体计算公式如下:

$$sim(d_i, d_j) = \frac{\sum_{k=1}^m (w_{ki} w_{kj})}{\sum_{k=1}^m W_{ki}^2 \sum_{k=1}^m w_{kj}^2} \quad (4)$$

式中, w_{ki} 表示第 i 篇文本的第 k 维属性权值.

3 文本聚类算法

3.1 蚁群文本聚类算法

蚁群文本聚类算法的思想是: 将待聚类的文本随机放置在二维网格空间中, 每个网格最多放置一个文本, 然后把一定数目蚂蚁置于网格空间中, 每只蚂蚁随机选择一个文本, 在网格空间中随机的移动, 并计算当前文本相似度, 通过转换概率选择下一个文本, 并进入到下一轮循环, 经过多次迭代, 所有文本按照其相似性聚集在一起.

蚂蚁在运动过程中抬起和放下文本的概率分别为 $P_{pick}(d_i)$ 和 $P_{down}(d_j)$, 它们计算公式分别为:

$$P_{pick}(d_i) = \left(\frac{k_1}{k_1 + f(d_i)}\right)^2 \quad (5)$$

$$P_{down}(d_i) = \begin{cases} 2f(d_i) & \text{if } (f(d_i) < k_2) \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

式中, k_1 和 k_2 是阈值常量.

对于文本聚类, 由于聚类的对象为文本向量, 文本群体相似度 $sim(d_i, d_j)$ 计算公式为:

$$f(d_i) = \sum_{d_j \in Neigh_{s \times s}(r)} \left| 1 - \frac{10(1 - sim(d_i, d_j))}{\alpha} \right| \quad (7)$$

$\alpha \in [1, 10]$

式中, $f(d_i)$ 表示群体相似度密度, $s \times s$ 表示单元 r 的邻域面积, α 表示群体相似系数.

3.2 改进蚁群文本聚类算法

为了解决文本聚类过程中蚂蚁移动的随机性和盲目性问题, 本文采用信息素指导蚂蚁移动的方向, 使蚂蚁在较少次的循环中完成文本聚类, 提高文本聚类效率和聚类结果精度.

在蚁群算法的路径搜索过程中, 每个蚂蚁都在其经过路径上释放信息素, 在选择下一条路口时, 信息量较大路径的被选择概率相对较大, 这样就形成蚁群算法的正反馈机制. 蚂蚁某一时刻状态可以表示为: (r, θ) , 其中 r 和 θ 分别表示蚂蚁所处的位置和移动方向, 蚂蚁从 (r, θ) 移动到 (r^*, θ^*) 由信息素浓度的响应函数和方位权重因子决定, 信息素浓度的响应函数计算公式为:

$$W(\sigma) = \left(1 + \frac{\sigma}{1 + \gamma\sigma}\right)^\beta \quad (8)$$

式中, σ 表示信息素, β 表示对信息素刺激的敏感程度.

当信息素浓度初始化为 $\sigma=0$, 蚂蚁每一次循环完成后, 在网格空间中留下一定浓度的信息素, 且信息素具有挥发性, 在下次移动之前的信息素浓度

为:

$$\sigma_r = \sigma_r + \left[\eta + \frac{n}{a} \right] \quad (9)$$

式中, a 表示蚂蚁个数, n 表示算法执行的次数, η 表示信息素浓度的挥发系数.

$\tilde{w}(\Delta\theta)$ 表示方位权重因子, 其中 $\Delta\theta$ 表示每一步蚂蚁在方位上的改变量, 这样蚂蚁处于平面上任一个网格位置, 可以感知各个方向的信息激素浓度, 根据信息激素浓度做出相应的响应. 设蚂蚁从网格 k 到 i 的转移概率为 p_{ki} , 由计算公式为:

$$p_{ki} = \frac{W(\sigma_i) \tilde{w}(\Delta\theta_i)}{\sum_{j/k} W(\sigma_j) \tilde{w}(\Delta\theta_j)} \quad (10)$$

式中, j/k 表示格 k 周围的八个相邻网格空间.

因此, 由于本文在二维网格空间中引入蚂蚁释放的信息素, 蚂蚁在每一次移动过程, 都会对周围八个网格空间中信息素浓度进行判断, 根据信息素浓度决定移动的方向, 这样有效避免了蚂蚁移动随机性和盲目性, 缩短了蚂蚁搜索时间.

3.3 改进蚁群算法的文本聚类流程

文本特征词向量集合作为蚁群算法的文本聚类算法的输入, 文本类别作为输出, 具体聚类流程:

- (1) 选择待聚类文本, 并进行预处理.
- (2) 提取文本的特征集.
- (3) 初始化蚁群文本聚类算法参数. 主要包括: 蚂蚁个数, 群体相似度, 算法迭代次数, 网格半径等.
- (4) 给蚂蚁赋初值, 将待聚类的文本向量随机放置在网格上.
- (5) 计算群体相似度和蚂蚁抬起或入下文本向量的概率.
- (6) 信息素的响应函数, 根据信息素计算转移概率, 从而决定蚂蚁下一步移动的方向.
- (7) 对路径上的信息素进行更新.
- (8) 达到最大迭代次数, 输入文本聚类结果.

改进蚁群算法的文本聚类具体流程如图 1 所示.

4 仿真实验

4.1 实验数据集

仿真实验在 CPU 为 Pentium 4 3.0GHz, RAM 为 1G, 操作系统为 WindowsXP. 实验数据集来自复旦大学中文文本分类语料库, 该语料库包括环境、计算机、交通、教育、法律、经济、能源、历史、体育、文学、医学等类别. 本文仿真实验只针对计算机、教育、经济、能源、医学等类别进行测试, 采用随机方法对

每个类别选择 500 篇文本进行聚类, 其中测试样本为 100 篇.

4.2 与标准蚁群文本聚类算法的效率比较

标准蚁群算法和改进蚁群算法的文本聚类时间比较如图 1 所示. 从图 1 可知, 改进蚁群算法的文本聚类平均执行时间要少于标准蚁群算法, 在聚类的文本数量较少时, 改进蚁群算法的优势不明显. 但是随着聚类的文本数量增加, 改进蚁群算法的文本聚类时间比标准蚁群算法耗费的时间越来越少, 当聚类文本的数量增加到 500, 标准蚁群算法需要 60 秒时间完成文本聚类过程, 然而改进蚁群算法只需要 35 秒就能完成文本聚类过程, 显示出了改进蚁群算法对大数量文本快速聚类的优势. 主要是由于改进蚁群算法通过信息素来指导蚂蚁下一步移动的方向, 将信息素和文本向量很好联系在一起, 使得改进蚁群文本聚类算法能在较少次的循环次数中完成文本聚类过程, 加快了文本聚类算法的收敛速度, 提高了文本聚类的效率.

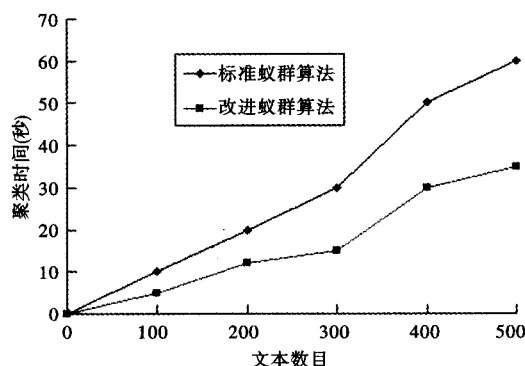


图 1 蚁群算法改进前后的文本聚类时间比较

4.3 与其他聚类算法的精度比较

采用 K-means、标准蚁群算法和本文改进蚁群算法对计算机、教育、经济、能源、医学等类别各 500 篇文本作为实验数据, 比较它们的聚类精度, 采用 F-Measure 作为算法的评价标准, F-Measure 定义如下:

$$F\text{-Measure} =$$

$$\sum_i \frac{n_i}{n} \max \left\{ \frac{2 \times \text{recall}(i, r) \times \text{precision}(i, r)}{\text{precision}(i, r) + \text{recall}(i, r)} \right\} \quad (11)$$

其中, $\text{recall}(i, r)$ 表示召回率, $\text{precision}(i, r)$ 表示准确率, 其定义如:

$$\text{recall}(i, r) = \frac{n(i, r)}{n_i} \quad (12)$$

$$\text{precision}(i, r) = \frac{n(i, r)}{n_r} \quad (13)$$

式中, n_i 表示预定义类别的个数, n_r 表示聚类形成的类别个数, $n(i, r)$ 表示聚类中包含 i 中的文本个数。

采用 K-means、标准蚁群算法和本文改进蚁群算法聚类的结果如图 2 所示。

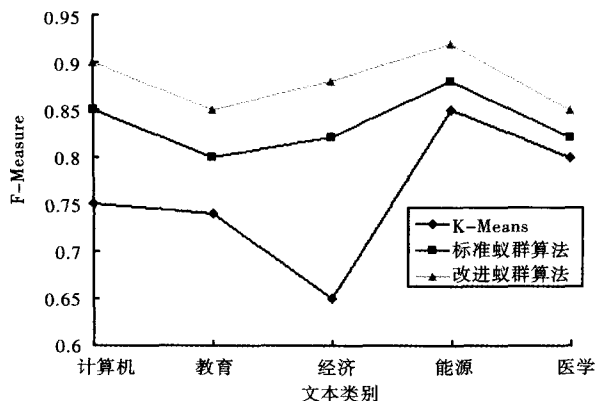


图 2 各种算法的文本聚类结果比较

从图 2 可知,在不同类别的文本集中,改进的蚁群算法明显优于其它算法,主要是因为 K-means 聚类算法对 K 值数目和初始划分比较敏感,当聚类数目或大或小时,文本聚类精度会受到影响;而标准蚁群文本聚类算法搜索具有随机性和盲目性,因此文本聚类精度比较低;改进蚁群文本聚类算法不需要预设聚类的数目,利用蚂蚁与蚂蚁、蚁群与环境的相互作用实现文本自动聚类,将一些主题相关的文本准确聚成一类,同时由于改进蚁群算法在信息素指导下,蚂蚁行动具有目标性,聚类时间明显少于标准蚁群文本聚类算法,提高了文本聚类的执行效率和文本聚类的精度,具有一定的优越性。

5 结束语

为提高文本聚类精度,文中提出一种改进的蚁群文本聚类算法,通过引入信息激素,指导蚂蚁始终向文本向量相对集中的区域移动,很好防止蚂蚁随机移动出现散点过多、收敛速度较慢等缺陷,使文本聚类速度加快。仿真实验结果表明,相对标准蚁群的文本聚类算法,在参数相同的条件下,改进的蚁群算法提高了文本聚类的效率,同时提高了文本聚类的精度。与其它文本聚类算法相比较,改进蚁群算法具有较明显优势,获得了更好的文本聚类效果。

参考文献:

- [1] 杨俊,陈贤富. 基于 KPCA 和 RBF 网络的文本分类研究[J]. 微电子学与计算机, 2011, 27(3): 122-125.
- [2] 刘远超,王晓龙,徐志明,等. 文档聚类综述[J]. 中文信息学报, 2006, 20(3): 55-63.
- [3] 任江涛. 一种用于文本聚类的改进的 K 均值算法[J]. 计算机应用, 2006, 26(6): 73-75.
- [4] 王国勇,徐建锁. TCBLSA: 一种中文文本聚类新方法[J]. 计算机工程, 2004(3): 21-22.
- [5] 庞剑锋,卜东波,白硕. 基于向量空间模型的文本自动分类系统的研究与实现[J]. 计算机应用研究, 2001(9): 23-26.
- [6] 吴斌,傅伟鹏,郑毅,等. 一种基于群体智能的 Web 文档聚类算法[J]. 计算机研究与发展, 2002, 39(11): 1429-1435.
- [7] 寇苏玲,蔡庆生. 中文文本分类中的特征选择研究[J]. 计算机仿真, 2007, 25(3): 100-104.

作者简介:

吕 岩 女, (1970-), 硕士, 副教授. 研究方向为计算机软件、数据库应用技术、信息安全。