

基于微粒群优化算法的文本模糊聚类方法

杜长海, 吉根林

(南京师范大学 数学与计算机科学学院, 江苏 南京 210097)

[摘要] 针对模糊 C-均值算法(FCM)具有局部最优问题和初值敏感性的缺陷, 将微粒群优化算法应用于文本模糊聚类, 提出了基于微粒群优化算法的模糊 C-均值算法 PFCM. 该算法首先采用实数编码方式对聚类原型进行编码, 利用微粒群优化算法的全局搜索性能对初始聚类原型的选取进行指导, 然后利用模糊 C-均值算法进行聚类. 使用算法 PFCM 对文本集合进行聚类实验, 并用目标函数值和划分系数来判断模糊划分的效果, 实验结果表明, 与 FCM 相比, 该算法具有较好的全局收敛性和较好的聚类结果.

[关键词] 模糊聚类, 微粒群优化, 模糊 C-均值, 文本聚类

[中图分类号] TP311 **[文献标识码]** A **[文章编号]** 1672-1292(2006)02-0030-04

Document Fuzzy Clustering Algorithm based on Particle Swarm Optimization Algorithm

DU Changhai, JI Genlin

(School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China)

Abstract: This paper analyzes local optimality and initialization dependence disadvantage of Fuzzy C-Means(FCM) and proposes an algorithm(PFCM)for document fuzzy clustering based on particle swarm optimization algorithm. Algorithm PFCM adopts real code for clustering prototype. Global searching of particle swarm optimization is used to instruct to choose clustering prototype and then clustering analysis is processed by FCM. This algorithm is used to conduct a clustering experiment on a document set. The quality of fuzzy partition is evaluated by objective function value and partition coefficient. The experimental results show that this algorithm can not only avoid local optima but also obtain better clustering result than FCM.

Key words: fuzzy clustering, particle swarm optimization, fuzzy C-means, text clustering

0 引言

在信息过载的今天, 文本聚类问题已经成为信息处理领域中的一项重要研究课题. 文本聚类是一种典型的无师机器学习问题, 其目标是将文本集合分成若干簇, 且同一簇内的文本相似度尽可能大, 而不同簇间的文本相似度尽可能小. 由于文本的多样性, 一个给定的文本往往可能归于多个类, 因此把模糊数学的思想应用于文本聚类, 得到文本属于各个类别的不确定程度, 从而更符合客观情况.

在各种模糊聚类算法中, 以模糊 C-均值(即 FCM)算法的应用较为广泛, 目前, 国内外学者已经把 FCM 算法用于文本模糊聚类研究^[1-3]. 但是, FCM 算法是一种局部搜索算法, 且具有初始聚类原型敏感性, 容易陷入局部最小值, 往往无法获得全局最优解. 因此, 如何克服 FCM 算法的局部搜索性和初值敏感性已成为亟待解决的问题.

微粒群优化算法(PSO)由美国社会心理学家 JAMES KENNEDY 和电气工程师 RUSSELL EBERHART 受鸟群觅食行为的启发于 1995 年共同提出^[4], 这是一种有效的基于群体智能理论的全局寻优算法. 系统

收稿日期: 2005-11-28.

基金项目: 江苏省自然科学基金资助项目(BK2005135).

作者简介: 杜长海(1977-), 硕士研究生, 主要从事数据挖掘技术的学习和研究. E-mail: duchanghai@tom.com

通讯联系人: 吉根林(1964-), 博士, 教授, 主要从事数据库与数据挖掘技术. E-mail: glji@njnu.edu.cn

初始化一组随机解,通过群体中微粒间的合作与竞争产生的群体智能指导迭代优化搜索.因此,本文将 PSO 算法与 FCM 算法相结合,提出了 PFCM 算法,并利用中文自然语言处理开放平台 www.nlp.org.cn 中文文本语料库中的部分文本作为测试集,建立基于名词的向量空间模型,对算法 PFCM 进行聚类实验,实验结果表明由该算法将 PSO 的全局搜索性和 FCM 的快速收敛性相结合,可以有效的克服 FCM 存在的问题,比 FCM 具有更好的聚类质量和综合性能.

1 基于名词的文本数字化建模

设待聚类的 n 个文本的集合为 $T = \{T_1, T_2, \dots, T_n\}$, x_i 为文本 T_i 的特征向量,每个文本有 m 个特征项, x_{ij} 为第 i 个文本中的第 j 个特征项的权重,则文本的特征向量空间模型为:

$$\begin{cases} X = \{x_1, x_2, \dots, x_n\} \\ x_i = (x_{i1}, x_{i2}, \dots, x_{im}), (i = 1, 2, \dots, n) \\ x_{ij} \in [0, 1], (i = 1, 2, \dots, n; j = 1, 2, \dots, m) \\ \sum_{j=1}^m x_{ij}^2 = 1, (i = 1, 2, \dots, n) \end{cases}$$

对于中文文本,词是最能够反映文本语义的基本单位,通常选择词作为特征项能充分表示文本的语义.但是,由于书面中文的词与词之间没有明显的切分标志,所以需要分词处理,通常可使用正向最大匹配算法.

由于名词对文本聚类的作用最大,因此本文选取名词作为特征项;然后合并名词中的同义词,如:“电脑”和“计算机”,“央视”和“中央电视台”;以总词库中的所有名词为初始特征集合,经过词频统计,有些名词在各个文本中都没有出现或都出现,这些特征项对于聚类没有作用,可以从初始特征集合中去掉,另外,还可把出现频率很低或很高的特征项去掉.则最终得到 m 个特征项的集合记为 $\{t_1, t_2, \dots, t_m\}$.

本文使用相对词频表示特征项在文本中的权重,其计算方法主要运用 TF-IDF^[5] 公式:

$$w_{ik} = tf_{ik} \cdot idf_k.$$

其中, tf_{ik} 表示特征项 t_k 在文本 T_i 中出现的频数, idf_k 表示特征项 t_k 的相反文本频数,较常用的计算公式为:

$$idf_k = \log\left(\frac{n}{N_k} + C\right).$$

其中, N_k 表示文本集合 T 中出现特征项 t_k 的文本数,常数 C 使得当 $N_k = n$ 时 idf_k 仍然大于 0,一般 C 取为一个较小的正常数,如 $C = 0.01$. 然后,进行词频均衡归一化处理,计算公式如下:

$$x_{ik} = \frac{\sqrt{tf_{ik} \cdot \log\left(\frac{n}{N_k} + C\right)}}{\sqrt{\sum_{k=1}^m tf_{ik} \cdot \log\left(\frac{n}{N_k} + C\right)}}$$

则形成文本 T_i 的特征向量 $x_i = (x_{i1}, x_{i2}, \dots, x_{im})$.

2 模糊 C-均值算法 FCM

设 n 个数据样本集合为 $X = \{x_1, x_2, \dots, x_n\} \subset R^m$, $x_k = (x_{k1}, x_{k2}, \dots, x_{km})^T (\in R^m)$ 为样本 x_k 的特征向量, m 为特征维数, x_{kj} 为特征向量 x_k 在第 j 维特征上的赋值; $c (2 \leq c \leq n)$ 是要将数据样本分成的类别数目; $p_i = (p_{i1}, p_{i2}, \dots, p_{im})^T (\in R^m, i = 1, \dots, c)$ 表示第 i 类的聚类原型,则 $P = (p_1, p_2, \dots, p_c) (\in R^{m \times c})$ 构成聚类原型矩阵; $U = (\mu_{ik})_{c \times n} (\in R^{c \times n})$ 是隶属度矩阵,其中 μ_{ik} 表示样本 x_k 对于聚类原型 p_i 的隶属程度. 则基于目标函数的模糊聚类分析可以用下式表达:

$$\begin{cases} \min J_b(U, P) = \sum_{k=1}^n \sum_{i=1}^c (\mu_{ik})^b (d_{ik})^2, b \in [1, \infty) \end{cases} \quad (1)$$

$$\begin{cases} \text{s. t. } \sum_{i=1}^c \mu_{ik} = 1, k = 1, 2, \dots, n \end{cases} \quad (2)$$

式中, d_{ik} 表示样本 x_k 与聚类原型 p_i 之间相异程度的度量, 通常可取为欧氏距离, 即 $d_{ik} = d(x_k, p_i) =$

$$\sqrt{\sum_{j=1}^m (x_{kj} - p_{ij})^2}.$$

FCM 算法就是寻找最佳的 U 和 P , 以使得该函数值 J_b 最小; 其算法流程如下:

(1) 给定聚类类别数 $c, 2 \leq c \leq n, n$ 是数据个数, 设定迭代停止阈值 ε , 初始化聚类原型模式 $P^{(0)}$, 设置迭代计数器 $t = 0$.

(2) 用下列公式计算和更新划分矩阵 $U^{(t)}$:

$$\text{对于 } \forall i, k, \text{ 如果 } \exists d_{ik}^{(t)} > 0, \text{ 则有 } \mu_{ik}^{(t)} = \left\{ \sum_{j=1}^c \left[\left(\frac{d_{ik}^{(t)}}{d_{jk}^{(t)}} \right)^{\frac{2}{b-1}} \right] \right\}^{-1}; \quad (3)$$

$$\text{如果 } \exists i, r, \text{ 使得 } d_{ir}^{(t)} = 0, \text{ 则有 } \mu_{ir}^{(t)} = 1, \text{ 且对 } j \neq r, \mu_{ij}^{(t)} = 0. \quad (4)$$

(3) 用公式(5)更新聚类原型矩阵 $P^{(t+1)}$:

$$p_i^{(t+1)} = \frac{\sum_{k=1}^n (\mu_{ik}^{(t+1)})^b \cdot x_k}{\sum_{k=1}^n (\mu_{ik}^{(t+1)})^b}, \quad i = 1, 2, \dots, c \quad (5)$$

(4) 如果 $\|P^{(t)} - P^{(t+1)}\| < \varepsilon$, 则算法停止并输出划分矩阵 U 和聚类原型 P , 否则令 $t = t + 1$, 转向(2). 其中, $\|\cdot\|$ 为某种合适的矩阵范数, 通常可取为矩阵 F -范数, 即 $\|P^{(t)} - P^{(t+1)}\|_F =$

$$\sqrt{\sum_{j=1}^m \sum_{i=1}^c (p_{ji}^{(t)} - p_{ji}^{(t+1)})^2}.$$

用公式(3)、(4)和(5)反复修改数据隶属度和聚类原型, 当算法收敛时, 理论上就得到了各类的聚类原型以及各个样本对于各聚类原型的隶属度, 从而完成模糊聚类划分. 尽管 FCM 有很高的搜索速度, 但 FCM 使一种局部搜索算法, 且对聚类原型的初值十分敏感, 如果初值选择不当, 它会收敛到局部极小点.

3 基于微粒群优化算法的模糊 C-均值算法 PFCM

3.1 基本微粒群优化算法

微粒群优化 (PSO) 算法是基于群体进化的算法, 具有记忆微粒最佳位置的能力和微粒间信息共享的机制, 通过种群间个体的合作与竞争来实现优化问题的求解. 采用速度—位置搜索模型, 将每一个可能产生的解表述为群体中的一个微粒, 每个微粒都具有自己的位置向量、速度向量及一个由目标函数决定的适应度. 所有微粒在搜索空间中以一定的速度飞行, 通过追随当前搜索到的最优值来寻找全局最优. 每一次迭代, 微粒(当前位置 present)通过动态跟踪两个极值来更新其速度和位置. 第一个是微粒从初始到当前迭代次数搜索产生的最优解: 个体极值 pBest. 第二个是微粒群目前的最优解: 全局极值 gBest. 微粒在解空间中不断跟踪个体极值与全局极值进行搜索, 直到达到规定的迭代次数或满足规定的误差标准为止.

基本 PSO 算法的数学表示如下^[6]: 设搜索空间为 m 维, 总微粒数为 N , 第 i 个微粒位置表示为向量 $X_i = (x_{i1}, x_{i2}, \dots, x_{im})$; 第 i 个微粒迄今为止搜索到的最优位置(对应于个体极值)为 $P_i = (p_{i1}, p_{i2}, \dots, p_{im})$, 整个微粒群迄今为止搜索到的最优位置(对应于全局极值)为 $P_g = (p_{g1}, p_{g2}, \dots, p_{gm})$, 第 i 个微粒的位置变化率(速度)为向量 $V_i = (v_{i1}, v_{i2}, \dots, v_{im})$, 每个微粒的位置按如下公式进行变化(“飞行”):

$$v_{ij}(t+1) = w \cdot v_{ij}(t) + c_1 \cdot \text{rand}() \cdot (p_{ij}(t) - x_{ij}(t)) + c_2 \cdot \text{rand}() \cdot (p_{gi}(t) - x_{ij}(t)) \quad (6)$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1), \quad 1 \leq i \leq N, \quad 1 \leq j \leq m \quad (7)$$

其中, c_1, c_2 称为加速常数, 代表将每个微粒推向 P_i 和 P_g 位置的统计加速项的权重; $\text{rand}()$ 为 $[0, 1]$ 之间均匀分布的随机数; w 称为惯性权重, 使微粒保持运动惯性, 使其有扩展搜索空间的趋势, 有能力探索新的区域. 第 j ($1 \leq j \leq m$) 维的位置变化范围为 $[X_{\min j}, X_{\max j}]$, 速度变化范围为 $[V_{\min j}, V_{\max j}]$ (即在迭代中若 v_{ij} 和 x_{ij} 超出了边界值, 将之设为边界值).

基本微粒群优化算法的流程图如图 1 所示.

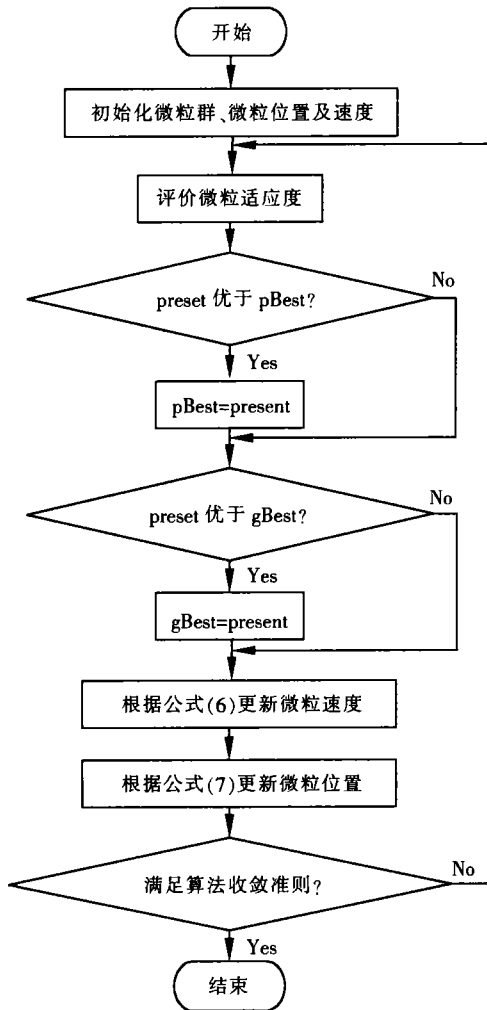


图1 基本 PSO 算法流程图

3.2 PFCM 算法

PFCM 算法是将基本 PSO 算法应用于 FCM 算法,可有效避免 FCM 算法对初始聚类原型的依赖,获取全局最优解,该算法由两个阶段组成,即 PSO 和 FCM. 第一阶段利用 PSO 算法的全局搜索性能,求得聚类原型,对精度不作要求,只要得到的聚类原型处于最优解的附近邻域即可;第二阶段以第一阶段得到的聚类原型作为初始聚类原型利用 FCM 算法精确求得最优聚类. PFCM 算法的流程如图 2 所示.

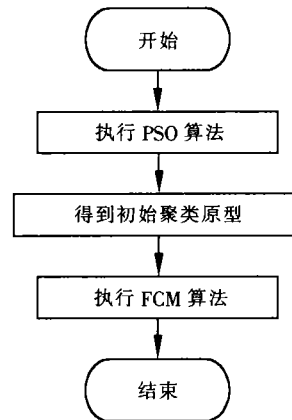


图2 PFCM 算法流程图

对于微粒的编码,本文采用基于聚类原型的实数编码方式,即每个微粒的位置由 c 个聚类原型组成. 由于数据样本的维数为 m ,所以微粒的位置是 $c \times m$ 维向量,同样微粒的速度也是 $c \times m$ 维向量. 此外,每个微粒还有一个适应度,计算公式为: $\text{fitness} = \frac{k}{J_0 + J_b}$ (k 是一个正常数, J_0 是一个较小的正常数).

可以看出,聚类原型越好,目标函数值越小,则该微粒的适应度越大. 同时需要记忆该微粒所搜索到的最佳位置及相应的最佳适应值.

4 实验及结果

从中文自然语言处理开放平台 (www.nlp.org.cn) 上下载中文文本语料库,选取 $c = 10$ 类(计算机类、艺术类、农业类、政治类、体育类、历史类、环境类、经济类、空间类和运输类) 各 50 篇共计 500 篇文本作为实验测试集.

实验参数分别为: $N = 40, c_1 = c_2 = 1.8, w = 0.8, G_{\max} = 100, \varepsilon = 0.00001, b = 2, k = 1.0, J_0 = 0.01$.

本文使用目标函数值 J_b 和文[7] 给出的聚类有效性函数——划分系数 $F(U; c)$ 来衡量 FCM 算法和 PFCM 算法的聚类结果有效性. 显然,目标函数值越小,其对应的聚类结果就越好. 划分系数 $F(U, c)$ 定义为 $F(U, c) = \frac{1}{n} \sum_{k=1}^n \sum_{i=1}^c (u_{ik})^2$; 可知,如果每个样本仅属于一类,即 u_{ik} 取 0 或 1,则聚类效果最好,此时, $F(U, c) = 1$, 因此,划分系数越大聚类效果越好. 实验结果如表 1 所示:

从表 1 可以看出, PFCM 算法的目标函数值(47.6438) 小于 FCM 算法的目标函数值(50.9834), PFCM 算法的划分系数(0.3482) 大于 FCM 算法的划分系数(0.2901), 表明 PFCM 算法比 FCM 算法的聚类结果更好,具有更好的全局寻优能力.

表1 聚类结果分析表

	FCM 算法	PFCM 算法
目标函数值	50.9834	47.6438
划分系数	0.2901	0.3482

(下转第 37 页)

4 结束语

低密度奇偶校验码是一种逼近香农限的线性分组码,是当前通信领域和数据处理方面的热门研究课题之一. LDPC 码的译码复杂度较低;但它的直接编码运算量较大,通常具有码长的二次方复杂度. 文章介绍了如何构造线性的编码,以简化 LDPC 码的编码运算量,并研究和设计了用大规模集成电路去实现一个 LDPC 码的编码. 以(6,2,3)码为例,采用基于半随机校验矩阵的编码方法,以控制编码运算量为线性复杂度,并在 QuartusII5.0 软件平台上采用基于 CPLD 的 Verilog HDL 语言编程仿真实现了有效编码的过程,给出了编码的结构图和仿真波形,为 LDPC 码的硬件实现和实际应用提供了依据.

[参考文献] (References)

- [1] MCKAY D J C. Good error-correcting codes based on very sparse matrices[J]. IEEE Trans Inform Theory, 1999,45(3):399-431.
- [2] BENJAMIN LEVINE, TAYLOR R REED, HERMAN SCHMIT. Implementation of near shannon limit error-correcting codes using reconfigurable hardware[C]//Field-Programmable Custom Computing Machines. IEEE Symposium, 2000:217-226.
- [3] FOSSORIER M. Iterative reliability-based decoding of low density parity check codes[J]. IEEE J Select Areas Commun, 2001,19(5):908-917.
- [4] 袁俊泉,孙敏琪,曹瑞. Verilog HDL 数字系统设计及其应用[M]. 西安:西安电子科技大学出版社,2002.
YUAN Junquan, SUN Minqi, CAO Rui. Verilog HDL Design and Applications of Digital System[M]. Xi'an: Xidian University Press,2002. (in Chinese)

[责任编辑:刘 健]

(上接第 33 页)

5 结束语

本文针对 FCM 算法存在的问题(局部搜索性和初始聚类原型敏感性)提出了基于 PSO 的 PFCM 算法,并应用于文本聚类实验,实验结果表明该算法是一种有效的方法. 只要文本特征抽取准确,并与其语义相结合,那么就可以提高 PFCM 算法的有效性. 但是文本聚类的难点之一是如何正确地提取文本特征,因为在分词中存在少量的语法歧义;难点之二是特征空间的高维性和特征向量的稀疏性. 因此,如何消除语法歧义、如何降低特征空间的维数和提高聚类的效率和精度,有待于今后进一步深入研究和完善.

[参考文献] (References)

- [1] KRISHNAPRAM R, JOSHI A, YI L. A fuzzy relative of the k -medoids algorithm with application to web document and snippet clustering[C]//Proc IEEE Intl Conf Fuzzy System-FUZZ IEEE. Seoul, 1999: 1281-1286.
- [2] KUMMAMURU K, DHAWALE A, KRISHNAPRAM R. Fuzzy co-clustering of documents and keywords[C]//IEEE Intl Conf Fuzzy System-FUZZ IEEE. St. Louis, Missouri, 2003: 772-777.
- [3] 林建敏,谢康林. 基于 PAT-array 和模糊聚类的文本聚类方法[J]. 计算机工程, 2004, 30(12): 126-127.
LIN Jianmin, XIE Kanglin. An approach of text clustering based on PAT-array and fuzzy clustering[J]. Computer Engineering, 2004, 30(12): 126-127. (in Chinese)
- [4] KENNEDY J, EBERHART R C. Particle swarm optimization[C]//Proc IEEE Int'l Conf on Neural Networks. Piscataway, NJ: IEEE Service Center, 1995: 1942-1948.
- [5] SALTON G. Introduction to Modern Information Retrieval[M]. New York: McGraw2Hill Book Company, 1983.
- [6] EBERHART R C, SHI Y. Particle swarm optimization: developments, applications and resources[C]//Proc Congress on Evolutionary Computation. Piscataway, NJ: IEEE Press, 2001: 81-86.
- [7] BEZDEK J C. Clustering validity with fuzzy sets[J]. J Mathematical Biology, 1974(1): 57-71.

[责任编辑:刘 健]