

基于混沌二进制粒子群优化的 KNN 文本分类算法

徐 辉

(广西财经学院 信息与统计学院, 广西 南宁 530003)

摘 要: 中文文本分类的主要问题是特征空间的高维性. 提出了基于混沌二进制粒子群的 KNN 文本分类算法, 利用混沌二进制粒子群算法遍历训练集的特征空间, 选择特征子空间, 然后在特征子空间中使用 KNN 算法进行文本分类. 在粒子群的迭代优化过程中, 利用混沌映射, 指导群体进行混沌搜索, 使算法摆脱局部最优, 扩大寻找全局最优解的能力. 实验结果表明, 提出的新分类算法对中文文本分类是有效的, 其分类准确率、召回率都优于 KNN 算法.

关键词: 二进制粒子群; 混沌; K 最近邻; 文本分类

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2012)08-0204-05

KNN Text Classification Algorithm Based on Chaotic Binary Particle Swarm Optimization

XU Hui

(School of Information & Statistics, Guangxi University of Finance & Economics, Nanning 530003, China)

Abstract: The main problem of Chinese text classification is the high dimensional feature space. Based on chaotic particle swarm optimization, KNN text classification algorithm is proposed. It uses chaotic particle swarm algorithm to traverse feature space of the training set, selects the feature subspace, and then it uses KNN algorithm to classify text in feature subspace. In particle swarm's iterative process, It uses chaotic map to guide swarms for chaotic search, it makes the algorithm out of local optimum, and expands the ability of finding global optimal solution. Experimental results show that the proposed new classification algorithm for Chinese text classification is effective, the classification accuracy and recall are better than KNN algorithm.

Key words: binary particle swarm; chaos; K-Nearest Neighbor(KNN); text classification

1 引言

文本分类是按照预先定义的一组类别, 根据文本的内容将文本自动归入到一个或多个预定义类别的过程. 由于互联网的迅速发展和广泛应用, 在线文本信息的急剧增长, 文本分类已经成为处理和组织文本数据的关键技术之一. 自动文本分类已经在网页组织和归类、主题标识、实时排序文档、信息检索、搜索引擎、邮件分类、信息过滤等领域得到应用. 近年来, 研究学者提出了各种不同的文本分类算法, 常用的分类算法有支持向量机(SVM)、K 最近邻

(KNN)、神经网络(NNet)、朴素贝叶斯(NB)、决策树等分类算法^[1-2], 还出现了基于遗传算法^[3]、粒子群算法^[4]等演化理论的文本分类算法.

KNN 算法直接从训练数据中学习得到分类模型, 该算法简单、容易实现, 分类有效, 成为最好的文本分类算法之一. 但是 KNN 算法存在分类时间长的缺点, 因为每个测试样本在分类时都要与所有训练样本逐一比较, 才能找出最相似的 K 个近邻, 尤其在文本分类时, 文本特征向量的维数往往是几千维甚至几万维, 需要耗费大量的计算时间. 很多学者对 KNN 文本分类算法进行了改进, 例如, 刘海峰等

收稿日期: 2012-01-09; **修回日期:** 2012-03-07

基金项目: 广西科技攻关与新产品试制项目(桂科攻 10123012-7)

人^[5]针对样本分布偏斜的文本集,提出了改进的互信息特征选择降维方法和改进的 KNN 文本分类方法.改进的 KNN 算法取得较好的分类效率.

由于文本数据的主要困难是特征空间的维数高和文本表示向量的稀疏性^[6],如何降低维数而又不损失分类性能,成为文本自动分类中最重要的问题.因此,本文提出一种不同的分类算法,新算法是利用混沌二进制粒子群优化(CBPSO)算法在训练文本集上选择最优的特征项构成特征子空间,并在特征子空间上实施 KNN 分类的文本分类新算法.

2 中文文本预处理

文本分类的过程主要由文本预处理、分类器设计和效果评估三个部分组成.由于中文文本几乎是无结构的、以计算机难以理解的自然语言表示的文本.因此,要完成中文文本的分类任务,首要工作是中文预处理,也就是将中文文本转换成分类算法能自动处理的、结构化的数字形式,这一过程包括中文分词、词频统计、文本向量表示、文本特征选择、建立 VSM 模型等关键技术.其次是确定文本所属的类别,通常利用统计方法或机器学习方法设计一个分类器,使用该分类器将新文本分到一个或多个类别.

2.1 中文分词

中文分词是中文文本预处理的第一步,它将中文文本切分为一系列词组.本文采用张华平博士开发的中文词性分析系统(ICTCLAS)进行中文分词、词性标注,根据构建的停用词表,删除训练集的停用词和低频词组,保留名词和动词,作为原始的特征项集合.

2.2 文本向量表示

文本表示主要采用 G. Salton 等人提出的向量空间模型(VSM).经过中文分词后,每篇文本表示为由特征项的权重组成的特征向量形式,即文本 D_i 表示为权重向量 $(w_{i1}, w_{i2}, \dots, w_{iM})$,其中 w_{ik} 为特征项 t_k 在文本 D_i 的权重,一般采用词频-逆文档频率(TF-IDF)公式(1)来计算,并作归一化处理,使 w_{ik} 的值在 $[0, 1]$ 内.

$$w_{ik} = \frac{tf_{ik} \times \log(N/n_k + 0.01)}{\sqrt{\sum_{t_k \in D_i} [tf_{ik} \times \log(N/n_k + 0.01)]^2}} \quad (1)$$

式中, tf_{ik} 是特征项 t_k 在训练文本 D_i 的词频, n_k 是训练集中包含特征项 t_k 的文本数目, N 是训练集的文本总数, M 是特征项的个数.

TF-IDF 公式用于评估特征项对一个文本的重

要程度,它表示特征项在文本的权重与特征项在文本中出现的次数成正比,而与训练集中包含该特征项的文本数目成反比.公式表明,如果某个特征项在一个文本出现的次数很高,而在其他文本中很少出现,那么该特征项能够很好地表示该类的特征,具有很好的分类能力.

2.3 文本相似度

两个文本 D_i 和 D_j 的相似度采用向量夹角余弦,其计算公式表示为

$$\text{sim}(D_i, D_j) = \frac{\sum_{k=1}^M w_{ik} \times w_{jk}}{\sqrt{(\sum_{k=1}^M w_{ik}^2)(\sum_{k=1}^M w_{jk}^2)}} \quad (2)$$

2.4 文本特征选择

文本特征选择的目标是将信息量小的特征项从特征空间中删除,降低特征空间的维数.典型的特征选择方法包括文档频率(DF)、互信息(MI)、信息增益(IG)和 χ^2 统计量(CHI)等方法. Y. Yang 等人^[2]对英文文本比较了上述多种特征选择算法的效果后,得出结论: χ^2 统计法表现最好,其分类准确率最高,其次是 IG 方法, DF 方法的效果与 IG、CHI 的效果大体相当,则 MI 方法的效果最差.代六玲等人^[6]针对中文文本比较了这 4 种特征选择方法的质量,实验表明,使用 SVM 和 KNN 分类器时, DF 比其他 3 种方法更优, MI 方法的性能不稳定.

χ^2 统计量衡量特征项 t_i 与类别 C_j 之间的关联程度. χ^2 值越大,特征项 t_i 与该类之间的相关性越大,携带的类别信息也越多;反之则越少.特征项 t_i 与类别 C_j 的 χ^2 统计量表示为式(3)^[2]:

$$\chi^2(t_i, C_j) = \frac{N(AD - BC)^2}{(A + C)(B + D)(A + B)(C + D)} \quad (3)$$

式中, N 为训练集的文本总数, A 表示 C_j 类包含 t_i 的文本个数, B 表示除 C_j 类的其他类中包含 t_i 的文本个数, C 表示 C_j 类中不包含 t_i 的文本个数, D 表示其他类中不包含 t_i 的文本个数.

对每一对 C_j 和 t_i , 计算 $\chi^2(t_i, C_j)$, 每类的特征项按照 χ^2 统计量的值降序排列, 根据特征项的总数、每类的文本个数和训练集总的文本个数, 确定每类选取的特征项个数, 据此选取 χ^2 值较高的特征项, 最终得到特征降维后的训练文本集的特征项集.

2.5 构建训练文本集的 VSM 模型

根据特征选择后得到的特征项集, 对训练集的每篇文本 D_i , 计算各特征项的 TF-IDF 值, 建立特

征项集组成的权重向量 $(w_{i1}, w_{i2}, \dots, w_{iM}), i=1, 2, \dots, N$.

3 混沌二进制粒子群优化算法

3.1 二进制粒子群优化算法

粒子群优化(PSO)算法是 Kennedy 和 Eberhart 通过对鸟群觅食过程的分析和模拟,于 1995 年提出的群体演化计算技术,随后又提出了解决组合优化问题的二进制粒子群算法(BPSO)^[7]. 在二进制空间中,粒子群由 d 维空间的 P 个粒子组成,每个粒子的位置由一串二进制位组成,第 i 个粒子的位置表示为向量 $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$,其中 x_{ij} 的值为二进制位 0 或 1,1 表示选择相应特征,0 表示未选择相应特征;粒子 i 的速度表示为向量 $V_i = (v_{i1}, v_{i2}, \dots, v_{id})$,初始速度为 $[0, 1]$ 内的随机小数,粒子的速度限制在 $[V_{\min}, V_{\max}]$ 内;向量 $Pb_i = (p_{i1}, p_{i2}, \dots, p_{id})$ 表示粒子 i 迄今为止搜索到的最优位置. 种群中所有粒子的最优位置,称为全局最优位置,记为 $Gb = (g_1, g_2, \dots, g_d)$. 在 BPSO 算法中,每次迭代时,每个粒子的速度和位置按照式(4)、(5)、(6)、(7)计算. 然后评估每个粒子的最优适应度值 $pbest$ 和全局最优适应度值 $gbest$.

$$v_{ij}^{t+1} = wv_{ij}^t + c_1 r_1 (p_{ij} - x_{ij}^t) + c_2 r_2 (g_j - x_{ij}^t) \quad (4)$$

$$v_{ij}^{t+1} = \begin{cases} v_{\max}, & v_{ij}^{t+1} > v_{\max} \\ v_{\min}, & v_{ij}^{t+1} < v_{\min} \end{cases} \quad (5)$$

$$s(v_{ij}^{t+1}) = \frac{1}{1 + e^{-v_{ij}^{t+1}}} \quad (6)$$

$$x_{ij}^{t+1} = \begin{cases} 1, & rand < s(v_{ij}^{t+1}) \\ 0, & rand \geq s(v_{ij}^{t+1}) \end{cases} \quad (7)$$

式(4)中, w 是惯性权重, c_1, c_2 是加速因子, $rand$, r_1, r_2 是 $[0, 1]$ 内的随机小数; v_{ij}^t 和 v_{ij}^{t+1} 是粒子 i 更新前后的速度; x_{ij}^t 和 x_{ij}^{t+1} 是粒子 i 更新前后的位置. 式(5)限制粒子 i 速度的每个维值在 $[V_{\min}, V_{\max}]$ 内. 式(6)确定粒子位置更新的概率. 式(7)确定下一次迭代中要更新的粒子. 如果 $s(v_{ij}^{t+1})$ 值大于 $(0, 1)$ 内随机产生的小数,则置 x_{ij}^{t+1} 值为 1, 即选择相应的特征 j 为下一次迭代时要更新的特征;否则置 x_{ij}^{t+1} 值为 0, 不选择相应的特征 j 为下一次迭代时更新的特征.

3.2 惯性权重的混沌映射

在 BPSO 算法中,惯性权重控制粒子群的全局

搜索和局部搜索的能力,大的惯性权重有利于全局搜索,而小的惯性权重有利于局部搜索. 惯性权重 w 是影响问题求解过程收敛的关键因素,极大地影响 BPSO 的搜索过程,从而影响分类的精确度. 由于 BPSO 算法容易陷入局部最优,导致过早收敛. 将混沌映射引入到 BPSO 算法,构成一个混沌二进制粒子群优化(CBPSO)算法,使得 CBPSO 算法既克服早熟收敛,保证搜索到全局最优值,又能保证搜索速度,使分类算法的性能更优.

混沌是由确定性方程得到非确定的随机运动状态. 混沌具有随机性、遍历性和规律性等特点. 因此,在 BPSO 算法的每次迭代过程中,使用混沌映射来确定惯性权重值. 惯性权重的值根据式(8)的 Logistic 映射^[8]计算得到:

$$w(t+1) = u \times w(t) \times (1 - w(t)), \quad t = 0, 1, 2, \dots \quad (8)$$

式中, $w(t)$ 为混沌序列,值在 $(0, 1)$ 内, u 为控制参数,当 $3.571\ 448 \leq u \leq 4$ 时, Logistic 映射处于混沌状态. 特别是 $u=4$ 时,处于完全混沌状态. 当惯性权重接近 1 时, CBPSO 算法强化全局最优的搜索能力;当惯性权重值接近 0 时,增强局部最优的搜索能力.

4 基于 CBPSO 的 KNN 文本分类算法

新算法的基本思路是: CBPSO 算法每次迭代时,产生 P 个粒子,根据每个粒子的位置值在特征空间集上选出相应的特征项,构成相应的特征项子集,然后在这些特征项子集上分别对测试文本执行 KNN 分类,得到 P 组 K 个近邻,比较所有粒子的适应度值,得到当前最优的粒子,即当前全局最相似的 K 个近邻. 经过 CBPSO 算法的不断迭代,最终找出适应度最高的 K 个最近邻文本,再按照 KNN 分类器的方法,计算出测试文本应属的类别.

4.1 粒子结构

每个粒子的位置表示为二进制位串 $X_i = (x_{i1}, x_{i2}, \dots, x_{id})$,代表特征空间上所选择的特征子集,其中 d 是训练文本集经过特征选择后的特征空间维数, x_{ij} 的值为 0 或 1,1 代表选择相应的特征,0 代表未选择的特征. 粒子速度是 d 维向量,其分量按上述式(4)、(5)计算.

4.2 粒子的适应度函数

每个粒子的适应度函数采用 KNN 文本分类有效性的评价标准 F_1 计算, F_1 的定义如式(9):

$$F_1 = 2pr/(p+r) \quad (9)$$

式中, p 和 r 分别是分类的准确率、召回率; F_1 是 p 和 r 的调和平均值, 它更接近于这两个值的较小者。因此, 只有 p 和 r 的值都很高时, F_1 值才很高。 F_1 值越大, 即粒子适应度值越大, 说明分类的质量越好。

4.3 基于 CBPSO 的 KNN 文本分类算法描述

基于 CBPSO 的 KNN 文本分类的步骤如下:

输入: 训练文本集 D , 测试文本 X , 训练文本总数 N , 最近邻的文本个数 K , 特征项总数 F , 粒子个数 P , 粒子速度的下限 $V_{\min}$ 和上限 $V_{\max}$, 最大迭代次数 MAXIter 。

输出: F_1 值最优的 K 个近邻和测试文本所属的类别。

(1) 按照 2.1 节, 将训练集的每篇文本进行中文分词, 去掉停用词, 计算词的频数和词的文档频数。

(2) 按照 2.4 节的 χ^2 统计法进行特征选择, 选出训练文本集的特征项集, 实现对文本特征空间的降维处理。按照 2.2 节和 2.5 节, 建立训练集的 VSM 模型。

(3) 对测试文本 X , 进行中文分词, 根据特征项集, 计算每个特征项的 TF-IDF 值, 建立 X 的 VSM 向量。

(4) 初始化粒子群。随机产生 P 个粒子的位置 X_i 、速度 V_i 、粒子最优位置 $pbest_i$ 和全局最优位置 $gbest$, 粒子位置的每位为 0 或 1, 速度为 $(0, 1)$ 内的随机数; 当前迭代次数 $t=0$ 。

(5) FOR $i=1$ to P

(5.1) 评价粒子 i 的适应度 $F(X_i)$ 。这里按照 KNN 分类方法计算测试文本 X 的 K 个近邻, 并计算分类评价标准 F_1 的值, 以 F_1 值作为粒子的适应度。以 F_1 值最高的粒子为全局最优的粒子。

(5.2) 根据式(8)计算惯性权重混沌序列。

(5.3) 根据式(4)、(5)、(6)、(7), 更新粒子的速度和位置。

(5.4) 计算粒子 i 的最优位置 $pbest_i$; 计算粒子群的全局最优位置 $gbest$ 。如果 $F(X_i) > F(pbest_i)$, 则 $pbest_i = X_i$ 。如果 $F(X_i) > F(gbest)$, 则 $gbest = X_i$ 。

END

(6) $t=t+1$

(7) 检查终止条件是否满足, 若满足, 则转(8); 否则转(5)。终止条件是迭代次数超过 MAXIter 。

(8) 根据 $gbest$ 对应的 K 个近邻, 计算测试文本 X 所属的类别, 输出 X 所属的类别。

5 实验结果与分析

5.1 实验数据集

本文的实验数据来源于搜狗实验室的中文文本分类语料库, 该语料库分为 9 个类别, 每个类别有 1 990 篇文档。对每个类随机抽取 300 篇文档作为本文的实验文档集, 共 2 700 篇文档。

5.2 实验设计与结果

文本分类通常使用分类准确率、召回率和 F_1 值来评估分类算法对每个类别的分类性能。为了说明本文算法的分类性能, 将它与传统的 KNN 文本分类算法进行比较。实验环境为 Windows XP, 内存 2 GB, CPU 为 Core I3 2.13 GHz, 采用 C# 2005 多线程编程技术, 采用 5-折交叉验证法, 即将实验文档集分为 5 组, 轮流将每组文档作为测试文档, 另 4 组文档为训练文档进行实验, 取 5 次实验结果的平均值作为算法性能效果的估计。CBPSO-KNN 算法的参数为 $c_1=c_2=2$, 粒子数 $P=30$, 粒子速度的下限和上限分别为 $-2, 2$, 混沌序列的初值 $u_0=0.48$, 最大迭代次数为 100。两个算法的参数 K 值分别取 30、45、60、80、100, 实验结果表明 K 取 60 时, F_1 值最优, 表 1 所示为特征项总数为 2 000, $K=60$ 的实验结果。

表 1 KNN 算法和 CBPSO-KNN 算法的比较结果 %

KNN				CBPSO-KNN			
类别	准确率	召回率	F1 值	类别	准确率	召回率	F1 值
健康	93.49	74.67	83.03	健康	90.58	86.16	88.31
教育	71.69	75.33	73.46	教育	82.46	88.29	85.28
体育	93.43	89.64	91.50	体育	94.57	93.33	93.95
招聘	63.49	78.67	70.27	招聘	79.57	81.53	80.54
文化	74.57	60.00	66.5	文化	77.30	71.98	74.55
旅游	80.79	82.67	81.72	旅游	86.74	84.00	85.35
财经	82.55	84.67	83.59	财经	83.16	89.05	86.00
军事	84.55	89.35	86.88	军事	86.53	93.33	89.80
IT	77.40	69.33	73.15	IT	85.49	76.59	80.80
平均	80.22	78.26	78.90	平均	85.16	84.92	84.95

为了验证不同的特征维数对分类效果的影响, 取特征维数分别为 500、1 000、2 000、3 000、4 000、5 000, 实验得到的 F_1 值分布如图 1 所示。对于 CBPSO-KNN 算法, 特征维数达到 2 000 时, F_1 值变化最为显著, 特征维数大于 2 000 时, F_1 值变化不大,

表明并不是特征维数越大,分类效果越好,反而会增加计算时间.

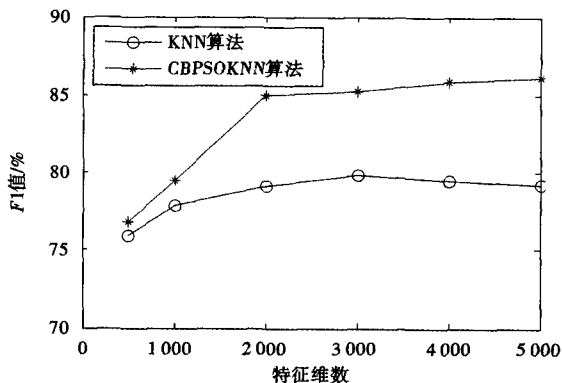


图1 特征维数对 F1 值的影响

分析表 1 和图 1 的实验结果,针对不同的特征维数,比较本文提出的 CBPSO-KNN 算法和传统的 KNN 算法,可以看到,对于大多数类别,在分类准确率、召回率和 F1 值方面,前者都比后者更优;从分类性能的平均值来看,前者也优于后者.

6 结束语

本文提出一种混沌二进制粒子群优化算法(CBPSO)和 KNN 算法相结合的文本分类新算法(CBPSO-KNN),利用混沌映射产生粒子群的惯性权重混沌序列,由 CBPSO 算法在训练集上随机选取特征项,生成若干个特征子集,在这些特征子集上构建 KNN 分类器,再由 CBPSO 算法进行寻优,从而找出最优的 K 近邻文本类别.以搜狗中文语料库进行实验,仿真实验结果表明,本文提出的 CBPSO-KNN 分类算法是有效的,不仅减少特征空间的维

数,而且提高了文本分类的精度.

参考文献:

- [1] 苏金树,张博锋,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] Yang Y, Pedersen J P. A comparative study on feature selection in text categorization[C] // The Fourteenth International Conference on Machine Learning. San Francisco, CA, USA: Morgan kaufmann Publishers Inc,1997:412-420.
- [3] 覃晓,元昌安,彭显忠,等. 基于词典和遗传算法的文本特征获取方法[J]. 计算机工程与设计,2008,29(21): 5651-5654.
- [4] 童亚拉,陈益. 一种基于混沌粒子群算法的网页分类规则抽取方法[J]. 微电子学与计算机,2009,26(2): 193-196.
- [5] 刘海峰,陈琦,刘守生,等. 一种基于数据偏斜的改进 KNN 文本分类[J]. 微电子学与计算机,2010,27(3): 51-53.
- [6] 代六玲,黄河燕,陈肇雄. 中文文本分类中特征抽取方法的比较研究[J]. 中文信息学报,2004,18(11): 26-32.
- [7] Kennedy J, Eberhart RC. A discrete binary version of the particle swarm algorithm[C] // IEEE Service Center, Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Piscataway, NJ: IEEE Press,1997:4104-4108.
- [8] 田雨波,潘朋朋. 混合粒子群算法优化神经网络的研究[J]. 微电子学与计算机,2011,28(6):156-159.

作者简介:

徐 辉 男,(1965—),副教授,研究方向为 Web 数据挖掘、文本挖掘.