

引用格式:李越颖. 基于邻域搜索的在线特征大数据分类方法[J]. 微电子学与计算机, 2021, 38(9): 61-66. [LI Y Y. Big data classification method of neighborhood search for online feature selection[J]. Microelectronics & Computer, 2021, 38(9): 61-66.]

基于邻域搜索的在线特征大数据分类方法

李越颖

(信阳农林学院 信息工程学院, 河南 信阳 464000)

摘 要: 针对现有算法在处理海量数据集时处理效率低的问题, 提出一种基于邻域搜索的在线特征选择(neighborhood search for online feature selection, NSOFS)并行大数据分类方法. 在 Map 阶段, 将大数据集进行分块, 针对动态未知特征空间, 通过萤火虫算法和模拟退火算法的优化, 对于在线特征进行邻域搜索, 选择最佳特征集, 将获得的特征集作为 Reduce 阶段输入特征, 然后使用内核支持向量机(Kernel Support Vector Machine, KSVM)对数据分类. 实验结果表明: 所提方法在精确率、召回率、F 值和时间等性能方面优于其他现有方法.

关键词: 在线特征选择; 邻域搜索; 模拟退火算法; 萤火虫算法; MapReduce

中图分类号: TP391.1

文献标识码: A

文章编号: 1000-7180(2021)09-0061-06

Big data classification method of neighborhood search for online feature selection

LI Yueying

(Xinyang Agriculture and Forestry University, Xinyang 464000, China)

Abstract: In order to solve the problem of low efficiency of existing algorithms when dealing with massive data sets, a parallel big data classification method is proposed based on neighborhood search for online feature selection. In the Map phase, the big data set is divided into splits. And the dynamic unknown feature space is optimized by the firefly algorithm and simulated annealing algorithm, the neighborhood search is carried out for the online features, then selecting the best feature. The obtained feature set is used as the input feature of the Reduce stage, and then the kernel support vector machine is used to classify the data. Experimental results show that the proposed method is superior to other existing methods in terms of accuracy, recall, F value and time.

Key words: online feature selection; neighborhood search; simulated annealing algorithm; firefly algorithm; MapReduce

1 引言

随着数字时代的发展, 使用数字数据量与日俱增, 越来越多的人来访问、创建和共享数据, 已经产生了大规模数据集, 这些数据集被定义为大数据, 从数据中挖掘有用信息是对数据研究的最终目的^[1-3], 其中分类是数据挖掘中的主要技术, 目前分类方法主要有神经网络、极限学习机、k 最近邻(kNN)和决

策树等方法^[4-6]. 对于常规数据分析方法, 由于可伸缩性限制, 这些方法并不直接适用于大数据, MapReduce 大数据框架提出的“分而治之”的方法, 使得传统方法应用于大数据分析^[7]. 大数据的最流行的分布式框架是 Apache Hadoop 和 Apache Spark, 最流行的方法之一是通过使用给定的数据集训练机器学习算法来构造分类模型, 常见的机器学习分类算法处理大规模数据时存在问题, 研究者已经

进行了相关的研究,如文献[8]提出的一种线性判别分析(Linear Discriminant Analysis, LDA)和支持向量机(Support Vector Machines, SVM)的大数据分类方法. 文献[9]提出的基于 MapReduce 改进 K-NN 的大数据分类算法研究. 文献[10]提出的二叉搜索树的 K-NN 分类算法. 以及文献[11]提出的基于 Spark 和 SimHash 的大数据 K-NN 分类算法,都提高了 kNN 分类方法的计算效率. 随着大数据分类方法的发展,有研究者开始针对不平衡大数据进行了研究, Lopez V 和 delRioS 等人将语言模糊规则的融合来解决大数据分类问题,并提出模糊分类系统,随后提出大数据的模糊决策树分类方法等. 但是针对更高维的海量数据,特征数过大,导致以上方法性能会下降,需要对特征数据进行降维^[12].

特征选择是常见的数据降维方法,但是特征选择针对的数据集特征已知且固定的,但是在实际大数据处理中,数据集特征经常是未知的,而在线特征选择针对的数据集可以是未知且不平衡的. 文献[13]提出一种高维不平衡数据的在线特征选择方法,设计了一种关于条件特征与决策类之间依存关系的有效在线特征选择框架,但该方法仅针对高维数据,并没有给出并行实现方法.

在现有大数据分类方法的基础上,提出一种基于模拟退火算法和萤火虫算法优化的邻域搜索在线特征选择的并行大数据分类方法. 该方法在 map 阶段将原始大数据集分解为样本块,然后使用模拟退火和萤火虫算法优化的 NSOFS 从大数据集中选择最优特征,将获得的部分结果合并为 reduce 阶段特征的最终向量,最后,使用 KSVM 分类器进行分类. 所提方法在高维度的数据集上分类性能优越.

2 在线特征选择

在线特征选择(Online feature selection, OFS)是一种特征选择算法,使用二值分类任务对 OFS 进行说明. 设 $\{(x_t, y_t) \mid t = 1, 2, \dots, T\}$ 是在训练过程中按顺序接收的训练数据实例序列,其中每个 $x_t \in R^d$ 是 d 维向量, $y_t \in \{1, -1\}$. 在线学习者会学习一个具有相同维数 $w \in R^d$ 的分类器,在线特征选择的设置中,需要选择 w 中相对较少的元素,并将其他元素设置为零,即,需要施加 $\|w\|_0 \leq B$ 约束,其中 B 是预定义的常数,因此 x 的最多 B 个特征将用于预测. 在 t 时,学习者接收到传入的示例 $x_t \in R^d$,然后基于其当前模型(即线性权重向量 w_t) 预测其类标签 $\hat{y}_t$.

$$\hat{y}_t = \text{sign}(w_t, x_t) \tag{1}$$

在进行预测后,真实标签 $y_t \in \{1, -1\}$ 将被揭示,然后学习者可以测量相对于 (x_t, y_t) 的损失 $l_t(w_t)$,表示预测结果和真实标签之间的差异. 在每次迭代结束时,学习者将根据一些学习规则更新权重向量 w_t .

OFS 最直接的方法之一是通过截断应用感知器算法,在每个步骤中,分类器首先通过线性权重向量 w_t 预测标签 $\hat{y}_t$. 如果 $\hat{y}_t$ 确,则 $w_{t+1} = w_t$; 否则,分类器将通过感知器规则更新 w_{t+1} ,以获得 $\hat{w}_{t+1} = w_t + \eta y_t x_t$,其中, η 表示学习率. 通过保持 $|\hat{w}_{t+1}|$ 中 B 个最大值,并将其余值设置为零,进一步截断该值. 用 w^B 表示的截断分类器将用于预测下一次预测.

3 基于邻域搜索的 OFS 大数据分类

提出一种邻域搜索的 OFS 大数据分类方法,该方法将萤火虫算法(FA)和模拟退火算法(SA)结合,提高领域搜索性能,选择佳特征集,最后使用内核支持向量机进行大数据分类.

在本文方法中,使用 Map 表示为大数据集样本矩阵, TempMap 表示临时映射的数据集矩阵.

映射函数				萤火虫位置			
Map ₁₁	Map ₁₂	...	Map _{1n}	TempMap ₁₍₁₎	TempMap ₁₍₂₎	...	TempMap _{1(n)}
Map ₂₁	Map ₂₂	...	Map _{2n}	TempMap ₂₍₁₎	TempMap ₂₍₂₎	...	TempMap _{2(n)}
⋮	⋮	...	⋮	⋮	⋮	...	⋮
Map _{m1}	Map _{m2}	...	Map _{mn}	TempMap _{m(1)}	TempMap _{m(2)}	...	TempMap _{m(n)}
Map矩阵				TempMap矩阵			

图 1 Map 矩阵与 TempMap 对应关系

图中可以看出,该矩阵的每一行显示了虚拟机上与该行相对应的数据集样本的序列,每一列显示大数据集样本分类或预测过程中的特征总和. 将大数据集样本矩阵对角线上的元素取为 100,这表示每个数据中心都在其内部进行萤火虫和模拟退火 100 次迭代的特征选择.

3.1 基于 FA 与 SA 的邻域搜索的 OFS

针对图 1 中两个大数据矩阵,使用两个不同的邻域搜索结构来选择最佳特征集. 为建立第一个搜索结构的邻域,FA 的移动运算符用于选择下一个最佳特征集;为建立第二个搜索结构的邻域,针对特征选择问题考虑了三个常见的运算符:交换、插入和逆运算,通过应用左移运算符,可以将特征集中的临时特征 F_{temp} 移至当前特征集 F_{curr} . 在 FA 中,最佳特征朝着最有吸引力的另一个特征的方向移动. 则最

佳特征的位置更新表达式如下所示:

$$X_i(t+1) = X_i(t) + \beta_0 e^{-\gamma_{ij}^2} (X_j(t) - X_i(t)) + \alpha \left(\text{rand} - \frac{1}{2} \right) \quad (2)$$

式中, $\beta_0 e^{-\gamma_{ij}^2}$ 是吸引力函数, γ 是光吸收系数, α 是随机参数, rand 产生是具有均匀分布的随机数, 取值区间 $[0, 1]$, r_{ij}^2 是两只萤火虫距离, 可表示为

$$r_{ij}^2 = \|x_i - x_j\|^2 = \sum_{k=1}^d (x_{ik} - x_{jk})^2 \quad (3)$$

x_{ik} 表示第 i 个萤火虫的第 k 个成分, 通过计算两个特征矩阵之间的距离, 可以找到 TempMap, 就可以借助分类指标来计算新的适应度值. 如果新适应度值低于当前适应度, 则更新位置, 另外, 将初始特征矩阵 F_{temp} 更新为当前特征矩阵 F_{curr} .

为了建立第二个搜索结构的邻域, 修改了经典的交换、插入和逆运算符. 在这些运算符中, 随机选择一个特征矩阵及其相应的数据集矩阵, 以相等的机会考虑数据集样本, 并对选定的特征矩阵实施运算符. 对于交换运算符: 两个现有数据集样本特征矩阵的位置相互交换; 对于插入运算符: 随机选择一个要素, 插入到选择的数据集矩阵位置; 对于逆运算符: 对于随机选择的两个特征之间的数据中心位置进行逆运算. 在每次迭代中, 算法从数据集样本矩阵 Map 中任意选择行和列, 并在对角线上没有等效元素的情况下将所选元素临时设置为零.

为解决 FA 中的全局最优问题, 选择 SA 以提高分类性能, SA 是一种在巨大搜索空间中估计全局优化的元启发式算法, 用于逼近给定函数的全局最优值, 在邻域搜索选择最佳特征子集时, 模拟退火 (SA) 中的参数 T 表示退火温度, B 表示玻尔兹曼常数和 r_{cool} 表示冷却速率. 从状态 s_0 开始, 一直保持到最大迭代次数 k_{max} , 或者直到获得包含最小能量 e_{min} 或更小的搜索为止.

在本文方法中, 呼叫邻居必须产生一个随意选择的具有指定状态的邻居, 随机调用 $[0, 1]$, 选择并返回序列中的一个值. 退火时间由温度 T 定义, 通过冷却速率 r_{cool} , 即 $T \leftarrow r_{\text{cool}} \times T$ 来更新温度, 并且算法返回到扰动步骤以创建新的运动. 当出现以下两种情况时, SA 收敛. 1) 对于温度 T , 解决方案在明显的迭代次数下不会增强; 2) 退火温度等于或低于地面温度 ($T \leq T_{\text{ground}}$), 即冻结状态.

考虑 T 是训练集, 而 m 是映射任务的数量. split 过程将 T 拆分为 m 个不相交的子集. 然后用 Map 函数处理 $T_i (i \in \{1, \dots, m\})$ 的每个子集. 由于此分区是连续执行的, 因此每个子集的样本数量大

致相同.

每个 T_i 上的 map 阶段都包含 OFS 算法, 在进行 FA 和 SA 的邻域搜索的 OFS, 选择最优特征向量 D , 对于每个映射任务的结果组成的二进制向量, 在 reduce 过程中求平均值, 得到:

$$x_{ij} = \frac{1}{m} \sum_{i=1}^m \text{res}_{ij}, j \in \{1, \dots, D\} \quad (4)$$

式中, res_{ij} 表示每个映射任务的结果, 其中 x_j 是 OFS 应用的比率, 其结果中包含特征 j , 该向量被称为完成 OFS 学习的结果, 并被用于构造在其他机器学习原理中使用的简化数据集. 借助 x_j 和阈值 θ , 将特征 $F = \{f_1, \dots, f_D\}$ 进行二值化操作, 目标是在计算特征向量后立即从原始数据集中去除非最优特征. 即

$$f_j = \begin{cases} 1, & x_j \geq \theta \\ 0, & x_j < \theta \end{cases} \quad (5)$$

式中, F 表示简化数据集选择的特征, θ 表示阈值, 是一个常数. 简化数据集的 MapReduce 范式以这种方式工作, 即, 每个 map 都处理一个实例并生成一个新的实例, 该实例仅包含在 F 中选择的特征. 最后, 要创建最终的简化数据集, 使用 reduce 阶段将生成的实例连接起来.

在 MapReduce 处理框架中, 首先对大数据集进行不同子集, 每个子集对应不同的 split, 然后在 map 进行特征选择, 输出键值对为 $\langle \text{分类准确率}, \text{特征} \rangle$, 对所有分类准确率进行排序, 选择最大的分类准确率, 其对应的特征即为最终特征, 所有特征以集合形式输入 reduce 阶段.

3.2 基于内核支持向量机的大数据分类

传统的分类算法处理大量数据效率低, 使用内核支持向量机 (Kernel Support Vector Machine, KSVM) 的大数据类方法, 能够在隐式以及高维特征空间中工作, 而这些空间不再需要计算该空间中的数据坐标. KSVM 的基本思想是: 在低维和高维的之间构造出一个函数, 满足在低维空间下对样本进行计算的结果与在高维中计算的结果是相近的. 在本文方法中使用正定核, 是 $\Phi(X_i)$ 和 $\Phi(X_j)$ 的点积, 可以使 $(y_i, x_i)_{i=1}^N, y_i \in \{-1, +1\}, x_i \in R^n$ 的指定标签训练数据表示正 (+1) 和负 (-1) 的类别标签. 数据点上的核 $k(X_i, X_j): R^n \times R^n \rightarrow R$ 正好是两个 $\Phi(X_i)$ 和 $\Phi(X_j)$ 的内积. 在高维特征空间中, 该算法识别出一个超平面, 该超平面通过最小化来精细分离数据, KSVM 的 dual 形式:

$$\begin{cases} \max W(a) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{ij} \alpha_i \alpha_j y_i y_j k(X_i, X_j) \\ s. t. \leq \alpha_i \leq C, \sum_{ij} \alpha_i y_j = 0 \end{cases}$$

(6)

式中, $C > 0$ 被称为正则化和约束违反之间的折衷. 决策函数表示为 $h(x) = \sum_{j=1}^m \alpha_j y_j k(X_i, X_j) + b$, 为了通过 KSVM 对一个点进行分类, 需要进行内核计算, 并且应该存储所有 m 个支持向量, 这会加快 KSVM 的分类. 对于特征向量 $X \in R_+^n$, 核函数为 $k(X_i, X_j), k(X_i, X_j) = \sum_{i=1}^n \min(X(i), X(j))$. 将核函数带入决策函数中, 可以得到

$$h(x) = \sum_{j=1}^m \alpha_j y_j (\sum_{i=1}^n \min(X_i, X_j)) + b$$

(7)

3.3 所提算法的 MapReduce 实现

本文提出一种基于模拟退火和萤火虫的邻域搜索的 OFS 并行大数据分类方法. 该方法在 map 阶段将原始大数据集分解为样本块, 然后使用模拟退火和多目标萤火虫优化算法从大数据集中选择最优特征, 将获得的部分结果合并为 reduce 阶段特征的最终向量, 最后, 使用 KSVM 分类器进行分类. 基于领域搜索的 OFS 如算法 1 所示.

算法 1: 基于领域搜索的 OFS 方法

- 1、来自原始数据集 $X = (x_{1j}, \dots, x_{nj})$ 的输入特征 $F, j = 1, \dots, M$, 定义吸光系数 γ
- 2、计算适应度函数 $f(x) =$ 分类准确度, TempMap $X = (x_{1j}, \dots, x_{nj})^T$
- 3、从数据集样本生成萤火虫的初始种群为 $x_i, i = 1, \dots, n, n = 100$
- 4、通过适应度函数 $f(x)$ 确定 x_i 处的光强度 I_i
- 5、while($t <$ 最大代数 < 100)
- 6、for $i = 1:n$
- 7、for $j = 1:i$
- 8、if $I_j > I_i$ and $F_{\text{curr}} > F_{\text{temp}}$
- 9、萤火虫 I 朝 d 维的 j 方向前进, 吸引力随距离 r 变化
- 10、else if
- 11、执行交换, 插入和逆运算, 转至 8
- 12、end
- 13、end
- 14、对萤火虫进行分类并找到当前的适应度值
- 15、if $t <$ 最大代数, 选择最佳特征
- 16、执行模拟退火算法
- 17、end while

KSVM 的并行大数据分类过程: 对于原始数据集 $X = (x_{1j}, \dots, x_{nj})$ 的输入特征 F , 首先执行 map 过程, 拆分大数据集, 然后根据算法 1 的 OFS 方法得

到 x_i . 之后执行 reduce 过程, 将 $\text{Shuffle}x_i = (\text{part}_0, \dots, \text{part}_n)$ 作为 KSVM 的输入, 进行预测.

4 实验结果与分析

为评价所提算法的性能, 对本文算法进行实验, 实验环境由 4 台集群结点组成, 选取 1 台作为主节点, 另外 3 台作为从节点, 每个节点的配置如下: Intel Core i7-12GB, RAM 为 64G, 硬盘容量为 500 G, 软件配置为 CentOS6. 5, Hadoop 版本 5. 4. 5, 4 个节点都连接在 100Mbps 的交换机上. 并在相同条件下与文献[9]中改进 K-NN 的大数据分类算法、文献[5]中基于改进蜻蜓算法的随机森林方法和文献[15]中基于蚁群-遗传优化的支持向量机方法进行比较. 本文选择 coverytype 数据集和 ECBDL14 数据集作为测试数据, 所用数据集的特性如表 1 所示.

表 1 实验数据集特征参数

数据集名称	测试样本	训练样本	特征	样本数 /split
coverytype	500 000	100 000	2 000	880
ECBDL14	65 003 913	2 897 917	631	1 894

表 2 二元问题的混淆矩阵

	预测阳性	预测阴性
阳性分类	真阳性(TP)	假阴性(FN)
阴性分类	假阳性(FP)	真阴性(TN)

针对不平衡分类性能的评估问题, 本文采用精确率、召回率、F 值来评估不平衡数据集上的分类性能. 实验将比较所提算法与其他 3 种算法在 2 个数据集进行的测试结果, 表 3 和表 4 分别给出了 coverytype 数据集和 ECBDL14 数据集的测试对比结果.

表 3 不同算法在 coverytype 数据集的对比结果

方法/性能	精确率	召回率	F 值	准确率
文献[9]	86. 33	89	87. 64	84. 32
文献[5]	84. 45	83. 13	84. 23	81. 09
文献[15]	88. 25	90	89. 16	86. 95
本文	91. 18	88. 13	89. 43	88. 12

表 4 不同算法在 ECBDL14 数据集的对比结果

方法/性能	精确率	召回率	F 值	准确率
文献[9]	76. 33	79	77. 64	80. 31
文献[5]	74. 45	75. 13	74. 79	75. 47
文献[15]	80. 25	82. 65	81. 43	83. 83
本文	87. 18	85. 13	86. 14	84. 09

从表 3 和表 4 的结果表明,所提算法在两个数据集上都取得了最好的指标性能,这是因为在所提算法中利用了 FA 和 SA 进行邻域搜索的在线特征选择,选择贡献度更大的特征,提高大数据集分类性能.而文献[9]中改进 K-NN 的大数据分类算法中对于大量的特征,使用了主成分分析方法进行降维,可能减去了贡献度大的特征,对分类性能有影响.文献[5]中基于改进蜻蜓算法的随机森林分类方法和文献[15]中基于蚁群-遗传优化的支持向量机方法的分类算法中在特征选择和分类方面都没有本文方法效果好,所以在性能上没有本文算法性能好,说明了本文算法具有更高的效率.

为了验证所提算法在时间方法的性能,将本文方法与文献[9]、文献[5]和文献[15]中方法进行比较,图 3 给出了不同数据量的运行时间比较.

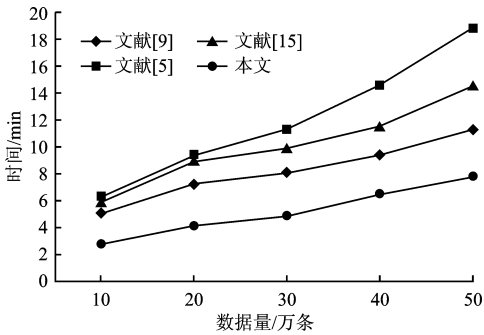


图 2 不同算法的时间性能

可以看出,所提方法在时间性能上优于其他三种方法,说明本文方法在时间性能的有效性.这是因为:(1)所提方法在 MapReduce 并行框架上运行,大大减少了花的时间;(2)大数据分类使用了 KSVM,将输入空间映射到高维特征空间,内核函数等于高维空间中内积,使得 SVM 不用计算复杂的非线性变换,大大降低了计算量,提高了计算效率和运行效率.

5 结束语

提出了一种邻域搜索的在线特征选择的大数据分类方法,用于解决大数据分类算法中存在的分类精度和时间的问题.在 Map 阶段开始将大数据集分解为子集,可以用更短的时间进行学习,然后使用 NSOFS 算法选择所需的特征,然后在 reduce 阶段将获得的部分特征输入到 KSVM 分类器进行分类.大数据集实验分析结果表明,所提算法能够以高性能对大数据进行分类,并且在保证准确率的同时,以较少的时间完成分类,并且在精确率、召回率、F 值

和准确率性能方面都要优于现有其他三种算法,而在用时却少于其他三种算法,说明本文算法的可行性和有效性.未来工作内容是研究扩大规模以挖掘稀疏数据和受限的优化问题.

参考文献:

[1] ZHANG Q C, YANG L T, CHEN Z K, et al. A survey on deep learning for big data[J]. Information Fusion, 2018 (42): 146-157. DOI: 10. 1016/j. infus. 2017. 10. 006.

[2] OUSSOUS A, BENJELLOUN F Z, LAHCEN A A, et al. Big data technologies; a survey[J]. Journal of King Saud University-Computer and Information Sciences, 2018, 30(4): 431-448. DOI: 10. 1016/j. jksuci. 2017. 06. 001.

[3] BEAM A L, KOHANE I S. Big data and machine learning in health care[J]. JAMA, 2018, 319(13): 1317-1318. DOI: 10. 1001/jama. 2017. 18391.

[4] 易明雨,肖赤心,潘晖,等. 用于大数据分类的快速隐层优化分布式极限学习机[J]. 计算机工程与应用, 2019, 55(16): 165-169. DOI: 10. 3778/j. issn. 1002-8331. 1804-0352.

YI M Y, XIAO C X, PAN H, et al. Fast hidden layer optimal extreme learning machine for big data classification[J]. Computer Engineering and Applications, 2019, 55(16): 165-169. DOI: 10. 3778/j. issn. 1002-8331. 1804-0352.

[5] LAKSHMANAPRABU S K, SHANKAR K, ILA-YARAJA M, et al. Random forest for big data classification in the internet of things using optimal features[J]. International Journal of Machine Learning and Cybernetics, 2019, 10(10): 2609-2618. DOI: 10. 1007/s13042-018-00916-z.

[6] 林倩瑜. 云服务环境下的大数据多标签属性分类技术[J]. 微电子学与计算机, 2019, 36(2): 101-104. DOI: 10. 19304/j. cnki. issn1000-7180. 2019. 02. 021.

LIN Q Y. Big data multi-label attribute classification technology in cloud service environment[J]. Microelectronics & Computer, 2019, 36(2): 101-104. DOI: 10. 19304/j. cnki. issn1000-7180. 2019. 02. 021.

[7] RAMÍREZ-GALLEGO S, FERNÁNDEZ A, GARCÍA S, et al. Big data: tutorial and guidelines on information and process fusion for analytics algorithms with MapReduce[J]. Information Fusion, 2018, 42: 51-61. DOI: 10. 1016/j. infus. 2017. 10. 001.

[8] VARATHARAJAN R, MANOGARAN G, PRIYAN M K. A big data classification approach using LDA with an enhanced SVM method for ECG signals in

- cloud computing[J]. *Multimedia Tools and Applications*, 2018, 77(8): 10195-10215. DOI: 10.1007/s11042-017-5318-1.
- [9] 李佳烨, 余浩. 基于 K 近邻的众包数据分类算法[J]. *计算机应用研究*, 2020, 37(4): 973-976. DOI: 10.19734/j.issn.1001-3695.2018.09.0736.
- LI J Y, YU H. Crowdsourcing data classification algorithm via K-nearest neighbor[J]. *Application Research of Computers*, 2020, 37(4): 973-976. DOI: 10.19734/j.issn.1001-3695.2018.09.0736.
- [10] HASSANAT A B A. Furthest-pair-based binary search tree for speeding big data classification using K-nearest neighbors[J]. *Big Data*, 2018, 6(3): 225-235. DOI: 10.1089/big.2018.0064.
- [11] 熊安萍, 蒋亚雄, 段杭彪, 等. 大数据环境下基于限定模糊规则的多分类器[J]. *计算机工程与设计*, 2019, 40(4): 1145-1150. DOI: 10.16208/j.issn1000-7024.2019.04.039.
- XIONG A P, JIANG Y X, DUAN H B, et al. Multi-classifier based on limited fuzzy rules in big data[J]. *Computer Engineering and Design*, 2019, 40(4): 1145-1150. DOI: 10.16208/j.issn1000-7024.2019.04.039.
- [12] KAUR D, AUJLA G S, KUMAR N, et al. Tensor-based big data management scheme for dimensionality reduction problem in smart grid systems: SDN perspective[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2018, 30(10): 1985-1998. DOI: 10.1109/TKDE.2018.2809747.
- [13] HU X G, ZHOU P, LI P P, et al. A survey on online feature selection with streaming features[J]. *Frontiers of Computer Science*, 2018, 12(3): 479-493. DOI: 10.1007/s11704-016-5489-3.
- [14] ZHOU P, HU X G, LI P P, et al. Online feature selection for high-dimensional class-imbalanced data[J]. *Knowledge-Based Systems*, 2017(136): 187-199. DOI: 10.1016/j.knosys.2017.09.006.
- [15] NURLAILY D, IRHAMAH, PURNAMI S W, et al. Support vector machine for imbalanced microarray dataset classification using ant colony optimization and genetic algorithm[J]. *AIP Conference Proceedings*, 2019, 2194(1): 020076. DOI: 10.1063/1.5139808.

作者简介:

李越颖 女, (1982-), 硕士, 讲师. 研究方向为模式识别、计算机网络技术. E-mail: 3169646479@qq.com.