

基于 MapReduce 改进 K-NN 的大数据分类算法研究

蒋 华, 韩 飞, 王 鑫, 王慧娇

(桂林电子科技大学 计算机与信息安全学院, 广西 桂林 541000)

摘 要: 针对传统 k-nearest neighbor algorithm(K-NN)分类算法计算量大、高维度海量数据集处理效率低的缺点, 本文基于 Hadoop 平台依托 MapReduce 分布式编程模型改写 Map 和 Reduce 函数, 并针对传统 K-NN 提出数据集主成分分析和临界区域数据预测时距离加权的方法. 首先, 对高维度数据进行主成分分析达到降维的目的, 从而提高运行效率; 其次, 在预测分类阶段加入完全区域和临界区域的概念, 临界区域对 k 个值 n 种类别进行距离加权, 提高准确率; 最后, 在 Hadoop 集群环境下的算法运行, 针对海量数据进一步提高其运行效率. 实验结果表明: 该算法在处理海量数据时极大地提高了计算效率和准确率.

关键词: 大数据; MapReduce; k-近邻算法; 临界区域; 主成分分析; 距离加权

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2018)10-0036-05

Big Data Classification Algorithm Based on MapReduce to Improve K-NN

JIANG Hua, HAN Fei, WANG Xin, WANG Hui-jiao

(School of Computer and Information Security, Guilin University of Electronic Technology, Guilin 541000, China)

Abstract: Aiming at the shortcomings of traditional k-nearest neighbor algorithm (K-NN) classification algorithm, such as large amount of calculation and high-dimension massive data set processing efficiency, this paper revises the Map and Reduce functions based on Hadoop platform by using MapReduce distributed programming model. Principal component analysis and critical region data when the distance weighted method. First, the principal component analysis of high-dimensional data to achieve the purpose of reducing dimension, so as to improve operational efficiency; secondly, in the classification stage of prediction, adding the concept of complete region and critical region, the critical region of k values of n species distance weighted. Finally, the algorithm running under the Hadoop cluster environment can further improve its operation efficiency against massive data. The experimental results show that this algorithm greatly improves the computational efficiency and accuracy when dealing with massive data

Key words: big data; MapReduce; K-NN; critical area; PCA; distance weighted

1 引言

在大数据的背景下, 如何在海量数据中挖掘有用数据已成为当前研究的热点. 其中, 分类是数据挖掘的主要技术之一^[1]. 目前常用的分类算法有神经网络、决策树、Bayes、SVM、K-NN、Boosting 等. 其

中 K-NN(k-nearest neighbor algorithm)分类算法作为一种简单、高效和无参数的数据分类算法, 然而, K-NN 需要大量的距离计算来分类相对较大的数据集, 并具有较大的内存开销. 分类需要很长时间, 在大数据处理问题上存在以下缺陷: (1) 由于大量的计算, 搜索最近的邻居 k 个样本需要计算每个

收稿日期: 2018-01-20; 修回日期: 2018-02-13

基金项目: 2016 广西高校中青年教师基础能力提升项目(ky2016YB150); 桂林电子科技大学研究生教育创新计划项目(2017YJCX48)

样本的训练集文本的距离,以便进行训练,这需要大量的计算时间;(2)该算法只计算“最近的”邻居样本.当一组数据量较大,样本之间的离散程度相对较大时,误差较大.

目前,有不少研究学者提出了关于对 K-NN 改进的算法,文献[2]通过基于熵的改进距离计算,基于加权距离的分类确定和绩效等级和定量转换来改进的 K-NN 算法克服了传统 K-NN 算法的缺点,该方法主要是人类的性能预测问题,对于大数据背景下海量数据的计算效率并没有那么大的优势;文献[3]提出基于 map-reduce 的分布式网格概略化 K-NN join(DSGMP-J)和基于 MapReduce 的 voronoi diagram 下 K-NN join(VDMP-J)两种方法提高 K-NN 连接效率和伸缩性,该方法极大的提高了 K-NN 连接效率,但是对于分类的准确率并没有大幅度提高;文献[4]通过聚类技术,发现了比原始数据集小得多的样本集,大大降低了 k 近邻算法的成本.但其分类精度取决于聚类效果.如果聚类结果不理想,分类精度不高.

本文提出一种基于 MapReduce 改进 K-NN 的大数据分类算法,在提高运行效率的同时提高分类的准确率.

2 相关理论

2.1 MapReduce

MapReduce^[5]是 Hadoop 的核心组部分,它是一种相对较新的并行计算形式,编程简单,可靠,支持可扩展性,和其他并行计算框架之间最大的区别之一是 MapReduce 同时执行并行和顺序计算:框架依次调用 map 和 reduce 函数,但在每个阶段中函数并行执行^[6]. MapReduce 算法由三个阶段组成.首先,执行地图功能.框架将相同的映射器任务 μ 分配给多个进程,但是这些进程将不同的输入作为 μ 的参数.映射器函数是无状态的,因为它没有关于以前处理的数据的任何信息,也不能与其他映射器任务进行通信.以这种方式,它只需要输入值来计算其输出值.这样,mapper 函数就可以并行运行,这样可以提高效率.一般来说,映射器 μ 采用单个字符串输入并生成任意数量的 $\langle \text{key}, \text{value} \rangle$ 对作为输出.其次,被称为洗牌或成组.在这个阶段,框架会自动按键排列映射器的输出.因此,这些值将按其键值进行分组.最后,洗牌处理完毕后,减速器会产生任意数量的 $\langle \text{key}, \text{value} \rangle$ 对作为输出^[7].像 mapper 任务一样,reducer 也是无状态的. MapReduce 框架的一个重要特性是所有映射

器任务必须在 Reducer 任务开始之前完成.该框架的这一方面支持顺序处理.但是,通过将不同的键分配给不同的进程, reducer 任务也支持并行化^[8]. MapReduce 处理流程如图 1 所示.

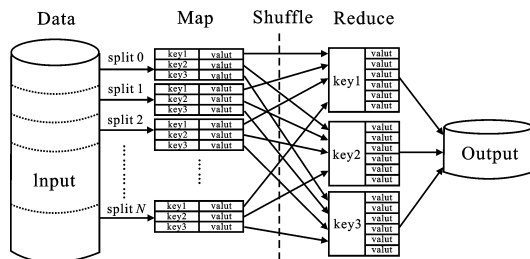


图 1 MapReduce 处理流程

2.2 主成分分析(PCA)

主成分分析是采用一种数学降维的数据处理方法^[9],是用来分析数据的多元统计分析中的一种方法,是用较少的特征来描述样本,以减少特征空间维数的方法,实际上是 K-L 转换. PCA 的原理是一个高维向量 x ,通过一个特征向量矩阵 U ,投影到一个低维向量空间,表征一个低维向量 y ,并且只丢失一些二次信息.换句话说,通过低维表示的向量和特征向量矩阵,可以重建对应的原始高维向量^[10].

PCA 方法步骤如下:

步骤 1:对原始数据进行标准化处理

假设样本数据矩阵为:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix} \quad (1)$$

那么按照如下方法对原始数据进行标准化处理:

$$x_{ij}^* = \frac{x_{ij} - \bar{x}}{\sqrt{\text{var}(x_j)}} \quad (2)$$

式中, $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_{ij}$, $\text{var}(x_i) = \frac{1}{n-1} \sum_{n=1}^n (x_{ij} - \bar{x})^2$, $i = 1, 2, \dots, n; j = 1, 2, \dots, p$.

步骤 2:计算样本相关系数矩阵

假设原始数据标准化后, X 表示归一化后的数据相关数为:

$$R = \begin{bmatrix} r_{11} & \cdots & r_{1p} \\ \vdots & \ddots & \vdots \\ r_{n1} & \cdots & r_{np} \end{bmatrix} \quad (3)$$

式中, $r_{ij} = \frac{\text{cov}(x_i, x_j)}{\sqrt{\text{var}(x_1)} \sqrt{\text{var}(x_2)}}$

$$\frac{\sum_{k=1}^{k=n} (x_{ki} - \bar{x}_i)(x_{kj} - \bar{x}_j)}{\sqrt{\sum_{k=1}^{k=n} (x_{ki} - \bar{x}_i)^2} \sqrt{\sum_{k=1}^{k=n} (x_{kj} - \bar{x}_j)^2}}, n > 1.$$

步骤 3: 计算相关系数矩阵 $\mathbf{R}$ 的特征值 ($\lambda_1, \lambda_2, \dots, \lambda_p$) 和相应的特征向量:

$$a_i = (a_{i1}, a_{i2}, \dots, a_{ip}), i = 1, 2, \dots, p \quad (4)$$

步骤 4: 选择主成分并写出主成分表达式

主成分分析可以得到 P 的一个主成分, 但是由于各主成分的方差在递减, 所以包含的信息量也在减少, 所以实际分析中, 一般不选择 P 主成分, 而是根据贡献率 各主成分的累积大小选择在 k 个主成分之前, 贡献率是主成分的方差占总方差的比例, 实际上也是总特征值的一定比例的组合, 即:

$$\text{贡献率} = \frac{\lambda_i}{\sum_{i=1}^p \lambda_i} \quad (5)$$

步骤 5: 计算主成分得分

根据标准原始数据, 根据每个样本, 可以获得主要组分的每个样本的新数据, 即主要组分得分. 据图形式如下:

$$\begin{pmatrix} F_{11} & \cdots & F_{1k} \\ \vdots & \ddots & \vdots \\ F_{n1} & \cdots & F_{nk} \end{pmatrix} \quad (6)$$

式中, $F_{ij} = a_{j1}x_{i1} + a_{j2}x_{i2} + \dots + a_{jp}x_{ip} (i = 1, 2, \dots, n; j = 1, 2, \dots, k)$.

步骤 6: 依据主成分得分的数据, 进一步对问题进行后续分析和建模

3 改进的 K-NN 算法

3.1 相关定义

由于传统 K-NN 算法在选取 k 个点中是以出现频率最高的那个类别作为预测分类的标准, 对离散程度较高的数据分类效果不佳. 因此, 本文为了提高分类的准确率在类别预测阶段提出完全区域和临界区域的概念, 并且在出现临界区域的情况下引入中心线的概念.

定义 1:

假设集合 A 和 B , 是由 n 维空间向量离散点组成, 并且 A 和 B 没有交集, 如图 2 所示.

现随机从 A 和 B 的并集中取出 k 个近邻点组成新的集合 K , 有如下 3 种情况:

情况 1: 如图 2, 如果集合 K 完全属于集合 A , 那么定义这 k 个点为集合 A 的完全区域, 记做: $\{1, 2,$

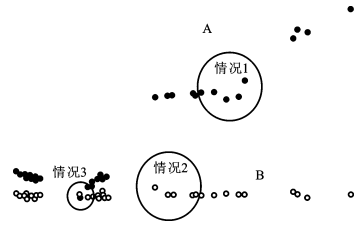


图 2 区域定义

$\dots, k\} \subset A$;

情况 2: 如图 2, 如果集合 K 完全属于集合 B , 那么定义这 k 个点为集合 B 的完全区域, 记做: $\{1, 2, \dots, k\} \subset B$;

情况 3: 如图 2, 如果这 k 个点中, 既有属于集合 A 的点也有属于集合 B 的点, 那么称这 k 个点所构成的区域为集合 A 和集合 B 的临界区域, 记做: $(A \cup K) \cup (B \cup K) = K$.

定义 2:

假设集合 A 是由 p 维空间的 n 个散点构成. 存在一条直线 l , 使这 n 个点到直线 l 的距离和最小, 那么称这条直线 l 为 n 个点的中心线, 如图 3.



图 3 中心线

3.2 K-NN 算法改进

K-NN^[11] 算法计算每个训练样本与数据集中的测试样本之间的距离, 然后返回 k 个最接近的样本, k 个样本出现频率最高的类作为测试数据的预测分类. 其中计算样本之间的距离主要使用欧几里得距离计算公式, 计算公式如公式 (7) 所示.

$$\text{dis}(P_1, P_2) = \sqrt{\sum_{i=1}^n (P_1 - P_2)^2} \quad (7)$$

为保证找到确切的 k 个最近邻, 它的时间复杂度 $O(n)$ 是线性的. 随着数据规模的不断增大, 时间复杂度不断增大, 算法的执行效率就越低. 并且针对离散程度较大的数据集分类效果不佳.

本文分别对多维的训练集和测试集进行主成分分析降维, 达到减少大量的计算提高分类效率的目的; 在算法预测分类阶段对区域进行分析, 针对临界区域引入中心线并进行距离加权, 在不影响分类效

率的情况下,提高分类精度.下面给出改进后算法基本思路:

输入:训练数据集 D_1 和样本数据集 $D_2, \{D_1, D_2\} \in D = \{X_i, Y_j\}$, X_i 表示样本属性值, Y_j 表示样本的类别;

输出:样本数据集的类别

算法步骤:

步骤 1:PCA 数据降维

在 Hadoop 平台环境下,基于 MapReduce 多维训练集和测试集主成分分析降维.将训练集 D_1 和测试集 D_2 按列组成 n 行 m 列矩阵 x ;每维的数据都减去该维的均值,使每一维的均值都为 0,这个过程称为特征中心化;求协方差矩阵的特征值及对应的特征向量;将特征向量按对应的特征值大小从上往下按行排列成矩阵,取前 k 行组成矩阵 p ; p 即为降维得到的 k 维数据.分别得到降维后的训练集 D_{k1} 和测试集 $D_{k2}, \{D_{k1}, D_{k2}\} \in D_k = \{X_{ki}, Y_{kj}\}$.

步骤 2:数据集计算

在 Hadoop 平台环境下,MapReduce 对样本数据进行分块处理,使用公式(7) 计算样本数据集与训练数据集属性之间的距离 $\text{Dis}\{X_{ki}, Y_{kj}\}$.

步骤 3: k 近邻选取

在 Hadoop 平台环境下,对计算得到的距离进行从小到大的排序,选取排在前面的 k 个训练集数据.

步骤 4:区域分析

在 Hadoop 平台环境下,统计 k 个训练数据的类别,进行类别分析并预测该测试数据的类别.

所选取的 k 个值的类别完全相同,为类别 T ,根据定义 1,该测试数据属于类别 T 的完全区域内.

所选取的 k 个值的类别种类大于等于 2,类别分别为 $T_1, T_2, \dots, T_n (n \leq K)$,根据定义 1,该预测数据应属于 n 个类别所对应的训练集的临界区域.

步骤 5:类别预测

根据步骤 4 得知该区域是属于哪种区域,如果该区域属于完全区域,那么类别预测为 T .如果该区域属于临界区域,对于该区域内的测试数据类别的预测,本文采用定义 2 的中心线和距离加权^[12]相结合的方法进行类别预测.根据定义 2,这 n 个类别的训练集 $\{D_{T1}, D_{T2}, \dots, D_{Tn}\}$ 都对应一条直线 $l = \{l_1, l_2, \dots, l_n\}$,使该类别的各个散点到直线 l 的距离和最小,分别计算求出该测试数据到直线 $l = \{l_1, l_2, \dots, l_n\}$ 的距离 $\text{Dis} = \{d_1, d_2, \dots, d_n\}$,则把 $\{d_1, d_2,$

$\dots, d_n\}$ 在 $\sum_{n=1}^n \text{Dis} = \sum_{n=1}^n \{d_1, d_2, \dots, d_n\}$ 中的比例的

倒数 $\{\frac{\sum_{n=1}^n \text{Dis}}{d_1}, \frac{\sum_{n=1}^n \text{Dis}}{d_2}, \dots, \frac{\sum_{n=1}^n \text{Dis}}{d_n}\}$ 作为距离权值, k

个值中的类别 T 的个数分别是 $C_1, C_2, \dots, C_n$, 计算

加权后的个数 $\{C_1 \frac{\sum_{n=1}^n \text{Dis}}{d_1}, C_2 \frac{\sum_{n=1}^n \text{Dis}}{d_2}, \dots, C_3 \frac{\sum_{n=1}^n \text{Dis}}{d_n}\}$

$\frac{\sum_{n=1}^n \text{Dis}}{d_n}\}$, $\{C_1 \frac{\sum_{n=1}^n \text{Dis}}{d_1}, C_2 \frac{\sum_{n=1}^n \text{Dis}}{d_2}, \dots, C_3 \frac{\sum_{n=1}^n \text{Dis}}{d_n}\}$ 最

大的作为该样本数据的预测类别,即为: $\text{Max}\{C_1$

$\frac{\sum_{n=1}^n \text{Dis}}{d_1}, C_2 \frac{\sum_{n=1}^n \text{Dis}}{d_2}, \dots, C_3 \frac{\sum_{n=1}^n \text{Dis}}{d_n}\}$ 的类别 T .

4 实验设计与结果分析

4.1 实验环境

实验集群规模 3 台电脑,1 台作为主节点(Master),2 台作为子节点(slaves),配置如下:

处理器为 Intel(R)Core(TM) i5-6300;内存为 32 GB;硬盘容量为 500 GB;系统版本为 Ubuntu 16.4;JDK 版本为 1.8,Hadoop 版本为 2.7.3.实验所用的开发环境为 Eclipse,并且安装了 MapReduce 插件,编程语言 Java.

本实验采用标准测试数据集 UCI 数据库的 CoverType Data.该数据集具有 54 个属性变量,58 万个样本,7 个类别,总大小 75.2 MB.本文实验从 58 万条数据中多组随机抽出部分作为训练集,随机抽出部分数据作为测试集,并保证训练集和测试集数据没有重复,具体情况如表 1.

表 1 实验数据选取

实验组别	训练集/条	测试集/条	备注
1	1 535	24 658	无重复
2	2 642	35 894	无重复
3	5 356	56 826	无重复
4	12 365	92 354	无重复
5	29 568	206 594	无重复
6	86 954	206 594	无重复
7	95 618	226 549	无重复
8	101 962	265 136	无重复
9	110 000	293 254	无重复
10	112 865	326 852	无重复

4.2 实验结果与分析

首先,给出本文方法在单机和 Hadoop 集群环境下以及传统 K-NN 分类算法在单机环境下,不同数据量的运行时间比较,如图 4 所示可以看出本文改进算法从时间耗用上是优于传统 K-NN 算法的,可以得出本文所提出的对于多维数据主成分分析进行降维的方法减少了该算法的计算量,从而优化了算法的运行效率. 本文基于 Hadoop 平台加上对原始数据主成分分析,弥补了传统 K-NN 算法针对海量数据巨大计算量的不足,降低了计算量,提高了计算效率和运行效率.

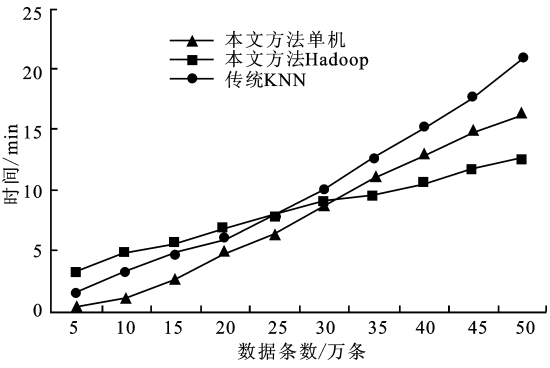


图 4 不同运行环境下的运行时间对比

其次,通过设定不同的 k 值进行试验,对比分析文献[4]LSC、传统 K-NN 和本文改进 K-NN 算法在不同 k 值下的分类准确率,如表 2 和图 5 所示.

表 2 取不同 k 值本文改进 K-NN 与传统 K-NN 分类对比情况

k 值	本文改进 K-NN / %	传统 K-NN / %	LSC / %
3	84.61	84.21	85.31
5	85.20	84.50	85.13
7	86.12	84.32	84.79
9	85.90	81.95	84.10
11	83.91	82.20	83.79
13	84.12	81.85	82.75
15	83.33	81.45	82.12
17	82.11	80.65	82.03
19	81.92	80.60	81.70

从图表中可以看出,传统 K-NN 取不同 k 值分类结果有很大的差别,当 k 值取 7 时其分类结果最理想达到 84.32%,随后随着 k 值的的增长分类准确率一直下降;LSC 算法随着 k 值的的增长基本成正比例下降,当 k 值取 3 时其聚类的效果最佳,所以也就是在 k 值取 3 时分类效果最理想,准确率为 85.31%;本文改进 K-NN 分类准确率先是随着 k

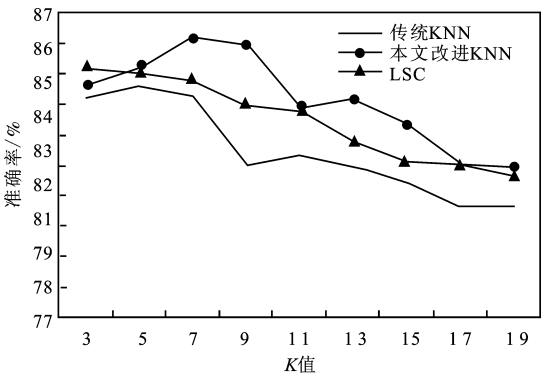


图 5 取不同 k 值本文改进 K-NN 与传统 K-NN 准确率对比

值的的增长而增长,当 k 值取 7 时达到顶峰 86.12%,然后呈波动势下降. 由图 5 种总体曲线分布可以看出,本文改进 K-NN 在 k 值的不同的对比,均优于传统 K-NN 算法;与 LSC 算法相比,本文改进 K-NN 虽然刚开始处于劣势,但是随着 k 值的的增长,本文改进 K-NN 分类效果基本均优于 LSC.

5 结束语

本文针对传统 K-NN 分类算法在处理海量数据集时计算量大以及高维度海量数据集处理效率低的缺点. 提出了对高维数据集进行主成分分析,并在传统 K-NN 算法中预测类别阶段加入完全区域和临界区域的概念,临界区域对 k 个值 n 种类别进行距离加权;再者,引入 MapReduce 并行计算模型,改写 Map 函数和 Reduce 函数使整个算法运行环境在 Hadoop 集群下. 通过大量数据集的实验分析验证,结果表明:该算法可以很好的对大量的高维数据进行分类,并且保证准确率的同时引用主成分分析和 MapReduce 编程模型,有效的提高了运行效率. 在此基础上,更深入地研究算法的鲁棒性,以及如何根据属性间的相关性计算出更加准确可靠的距离权值提高算法的准去率是今后主要研究的方向.

参考文献:

[1] 杨秀荣. 并行数据库中异常数据优化分类挖掘方法研究[J]. 微电子学与计算机, 2015,33(10):125-128.

[2] Li N, Kong H, Ma Y, et al. Human performance modeling for manufacturing based on an improved KNN algorithm[J]. International Journal of Advanced Manufacturing Technology, 2016, 84(1/4):473-483.

[3] 戴健, 丁治明. 基于 MapReduce 快速 kNN Join 方法[J]. 计算机学报, 2015, 38(1):99-108.

从实验结果可以看出本文的识别算法相对于文献中算法,识别率有所提高,拒识率也有所降低,说明本文算法对手写字符中的大小写字母和数字识别的通用性更好.

5 结束语

本文详细阐述了基于背景域串行分类器的脱机手写数字字符识别算法.这是一种对手写大小写字母和数字都通用的算法,且在识别率和拒识率上都达到理想的效果,具有可观的发展前景.本文交截特征中的阈值 δ 值暂时还是手动选取,有待进一步研究改进.

参考文献:

[1] 何斌,马天予,王运坚等. Visual C++数字图像处理(2版)[M]. 北京:人民邮电出版社. 2002.

[2] 赵锋,薛惠锋,王伟,等. 采用多级分类器的手写数字识别技术研究[J]. 微电子学与计算机. 2006, 23(12): 30-32.

[3] 汤茂斌,谢渝平,李就好. 基于神经网络算法的字符识

别方法研究[J]. 微电子学与计算机. 2009, 26(8): 91-47.

[4] 龚才春,刘荣兴. 基于整体特征的快速手写体数字字符识别[J]. 计算机工程与应用. 2004, 19(2): 82-83.

[5] 彭敏. 基于多级神经网络分类器的手写数字识别[D]. 沈阳: 沈阳工业大学, 2014.

[6] 姚超,卢朝阳,李静,等. 一种分类器级联的手写相似汉字识别方法[J]. 西安电子科技大学学报:自然科学版. 2015. 42(5): 26-31.

[7] 杜彦蕊,李珍,宋伟宏. 基于特征编码的手写字符识别技术[J]. 计算机工程. 2004. 30(5): 156-158.

[8] 黄瀚敏,汪先矩,易正俊,等. 一种基于特征提取的手写字符识别技术[J]. 重庆大学学报:自然科学版. 2000, 23(1): 66-69.

作者简介:

罗 佳 女, (1981-), 硕士, 副教授. 研究方向为图像处理. E-mail: luojia_dou@163. com.

刘大刚 男, (1979-), 博士, 副教授. 研究方向为地下工程.

杨 姝 女, (1981-), 硕士, 副教授. 研究方向为图像处理.

杨菊英 女, (1980-), 硕士, 讲师. 研究方向图像处理.

(上接第 40 页)

[4] 苏毅娟,邓振云,程德波,等. 大数据下的快速 KNN 分类算法[J]. 计算机应用研究, 2016, 33(4): 1003-1006.

[5] Dean J, Ghemawat S. MapReduce: a flexible data processing tool[J]. Communications of the Acm, 2010, 53(1): 72-77.

[6] Wang S, Wang H J, Qin X P, et al. Architecting big data: challenges, studies and forecasts[J]. Chinese Journal of Computers, 2011, 34(10): 1741-1752.

[7] 蒋华,胡莹. 基于 MapReduce 技术的 Argo 浮标剖面信息融合算法[J]. 计算机应用, 2015, 35(12): 3403-3407.

[8] He B, Fang W, Luo Q, et al. Mars: a map reduce framework on graphics processors[C]// International Conference on Parallel Architectures and Compilation Techniques. Minneapolis , USA, IEEE, 2017: 260-269.

[9] Schmitt E, Vakili K. The FastHCS algorithm for robust PCA[J]. Statistics & Computing, 2015, 26(6): 1-14.

[10] 周英,旧金武,卞月青. 大数据挖掘:系统方法与实例分析[M]. 北京:机械工业出版社, 2016.

[11] Zhang M L, Zhou Z H. ML-KNN: A lazy learning approach to multi-label learning[J]. Pattern Recognition, 2007, 40(7): 2038-2048.

[12] Jiang L, Cai Z, Wang D, et al. Bayesian Citation-KNN with distance weighting[J]. International Journal of Machine Learning & Cybernetics, 2014, 5(2): 193-199.

作者简介:

蒋 华 男, (1963-), 博士, 教授. 研究方向为信息安全.

韩 飞(通讯作者) 男, (1990-), 硕士. 研究方向为信息安全、数据挖掘、海洋大数据. E-mail: 1130195868@qq. com.

王 鑫 男, (1976-), 硕士, 副教授. 研究方向为无线传感网络.

王慧娇 女, (1976-), 硕士, 副教授. 研究方向为无线传感器网络.