计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2019,28(5):220−225 [doi: 10.15888/j.cnki.csa.006927]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于改进随机森林算法的文本分类研究与应用①

刘　勇, 兴艳云

(青岛科技大学 信息科学技术学院, 青岛 266061)

摘　要: 传统随机森林分类算法采用平均多数投票规则不能区分强弱分类器, 而且算法中超参数的取值需要调节优化. 在研究了随机森林算法在文本分类中的应用技术及其优缺点的基础上对其进行改进, 一方面对投票方法进行优化, 结合决策树的分类效果和预测概率进行加权投票, 另一方面提出一种结合随机搜索和网格搜索的算法对超参数调节优化. Python 环境下的实验结果表明本文方法在文本分类上具有良好的性能.

关键词: 随机森林; 文本分类; 加权投票; 超参数优化; 随机搜索; 网格搜索

引用格式: 刘勇,兴艳云.基于改进随机森林算法的文本分类研究与应用.计算机系统应用,2019,28(5):220–225. http://www.c-s-a.org.cn/1003-3254/6927.html

Research and Application of Text Classification Based on Improved Random Forest Algorithm

LIU Yong, XING Yan-Yun

(Information Science and Technology Academy, Qingdao University of Science and Technology, Qingdao 266061, China)

Abstract: Traditional random forest classification algorithm cannot distinguish the strong and weak classifiers by using the majority voting rule, and the value of its hyperparameter needs to be adjusted and optimized. This work studies the application technology of random forest algorithm in text classification and its advantages and disadvantages, and optimizes it. On one hand, optimize the voting method, perform weighted voting by combining classification effect and prediction probability of decision tree. On the other hand, an algorithm combining random search and grid search is proposed to optimize the hyperparameters in random forest. The experimental results in python environment show that the proposed method has sound performance in text classification.

Key words: random forest; text classification; weighted voting; hyperparametric optimization; random search; grid searchs

1　引言

目前, 随机森林算法是常用的机器学习方法, 在新闻分类、入侵检测[1]、内容信息过滤[2]、情感分析[3]、图像处理[4,5]等领域都有着广泛的应用, 具有很高的研究价值. 许多学者提出了随机森林的改进方法, 杨宏宇等采用 IG 算法和 ReliefF 算法对特征属性选择进行优化, 提出改进随机森林 (IRF, Improved Random Forest) 算法[6]. Abellán J 等提出基于不精确的信息增益分裂规则的信用随机森林 (CRF, Credal Random Forest) 算法[7]. Paul A 等提出一种迭代减少不重要特征和减少决策树数量的随机森林改进方法[8]. 对于随机森林在文本分类上的应用, 何珑针对数据集的不平衡问题, 通过改变样本抽样和赋予权重来改进算法[9]. 贺捷从加权投票和解决平局现象两方面进行随机森林分类算法改进[10]. 田宝明等结合基于词的和基于 LDA 主题的两种文本表示方法改进随机森林, 提高文本分类的性能[11].

随机森林忽视强弱分类器的差异以及超参数的优化是被广泛讨论和需要更好解决的两个问题. 本文采用基于决策树分类效果和预测概率的加权投票机制以

① 收稿时间: 2018-11-23; 修改时间: 2018-12-12; 采用时间: 2019-01-08; csa 在线出版时间: 2019-05-01

及结合随机搜索和网格搜索的超参数优化算法对随机森林算法进行改进, 提出用于文本分类的改进随机森林分类模型, 首先对文本进行空间向量表示和特征提取, 之后使用加权的随机森林进行分类, 并且对超参数进行调节优化, 最后对分类效果进行评估.

2 随机森林算法

2.1 算法介绍

随机森林算法[12]是一种基于 Bagging 的集成学习方法, 可以用于解决分类和回归问题, 本文就随机森林算法用于分类问题进行研究. 随机森林算法的基本构成单元是充分生长、没有剪枝的决策树. 算法流程包括生成随机森林和进行决策两部分, 如图 1 所示.

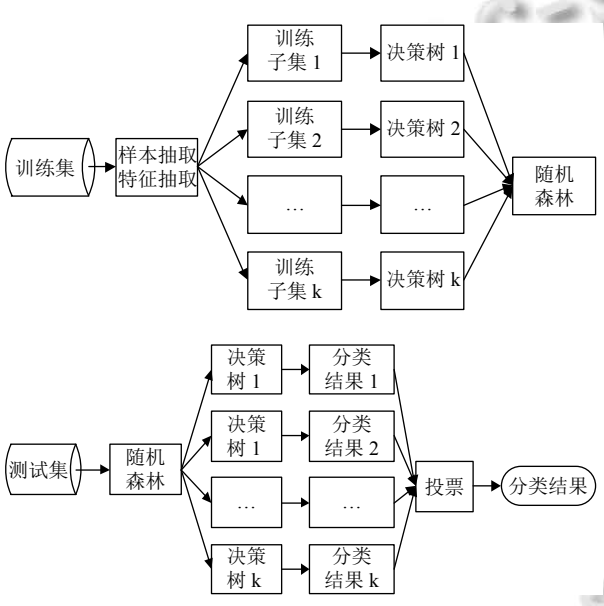

图 1 随机森林的算法流程

具体的算法步骤如下:

(1) 记给定原始训练集中的样本数量为 N, 特征属性数量为 M. 采用 bootstrap 抽样技术从原始训练集中抽取 N 个样本形成训练子集.

(2) 从 M 个特征属性中随机选择 m 个特征作为候选特征 ($m \le M$), 在决策树的每个节点按照某种规则 (基尼指数、信息增益率等) 选择最优属性进行分裂, 直到该节点的所有训练样例都属于同一类, 过程中完全分裂不剪枝.

(3) 重复上述两个步骤 k 次, 构建 k 棵决策树, 生成随机森林.

(4) 使用随机森林进行决策, 设 x 代表测试样本, h_i 代表单棵决策树, Y 代表输出变量即分类标签, I 为指示性函数, H 为随机森林模型, 决策公式为:

$$H(x) = \arg\max_{Y} \sum_{i=1}^{k} I(h_i(x) = Y) \quad (1)$$

即汇总每棵决策树对测试样本的分类结果, 得票数最多的类为最后的分类结果.

2.2 算法分析

随机森林算法采用的抽样方法 bootstrap 是一种有放回的简单随机抽样, 每个训练子集的抽取过程中都有约 37% 样本没有被抽中, 这些样本被称为袋外数据. 袋外数据具有很高的研究价值, Breiman 指出袋外数据估计是随机森林泛化误差的无偏估计, 可以替代数据集的交叉验证法. 袋外数据还可以用于检验每棵树分类效果的好坏[13]和估计随机森林算法中的超参数[14].

随机森林算法具有很多优点. 作为一种分类器组合算法, 其通过覆盖优化手段将若干弱分类器的能力进行综合, 使分类系统的总体性能得到优化, 比单个算法要好[15]. 在生成随机森林时, 每棵决策树相互独立、同时生成, 训练速度快并且容易做成并行化方法. 在选择样本时随机抽样以及在构建决策树时随机选择特征, 算法不容易陷入过拟合, 并且具有一定的抗噪能力.

与此同时, 随机森林算法也存在着不足之处. 随机森林算法在进行决策时采用多数投票算法, 不考虑强分类器和弱分类器的差异, 对于整体的分类结果有所影响. 算法中有很多超参数, 如决策树的棵数、节点分裂时参与判断的最大特征数、树的最大深度以及叶节点的最小样本数等. 它们需要在训练开始之前设置值, 但使用者无法控制模型内部的运行, 只能在不同的参数之间进行尝试, 如何对超参数进行调节取值对随机森林算法的性能有着很大的影响.

3 随机森林算法的改进

投票决策过程决定了最终的分类结果, 超参数的设置对于算法性能也至关重要. 本文在这两个方面提出以下改进.

3.1 加权投票方法

传统随机森林算法进行分类决策时, 采用平均多数投票法, 每一棵决策树输出自己的分类标签, 最终的分类结果为输出最多的类. 但是随机森林中的决策树分类效果不同, 有的决策树分类效果好, 有的决策树分类效果差. 按照平均多数投票法, 每棵决策树具有相同

的投票权重, 不能充分利用分类效果好的决策树的能力以及减少分类效果差的决策树对分类结果的负面影响.

本文设计一种结合决策树分类效果和类概率的加权投票方法用于随机森林的决策阶段. 决策树的分类效果通过袋外数据的分类正确率来体现, 分类正确率高的决策树分类效果好, 赋予的权重高. 设总样本数为 X, 正确分类的样本数为 X^{correct}, 随机森林中决策树 i 的权重为:

$$weight(i)=\frac{X^{\text{correct}}(i)}{X} \tag{2}$$

对于每一个测试样本, 决策树输出样本属于各个类的预测概率 p. 在投票阶段结合决策树的权重, 得到样本属于类 Y 的加权值:

$$Z(Y)=\sum_{i=1}^{k} weight(i)^{*} p_{Y}(i) \tag{3}$$

之后选择值最大的类为分类结果.

改进后的随机森林分类算法模型包括训练模块和检测模块两部分, 算法流程如图 2 所示.

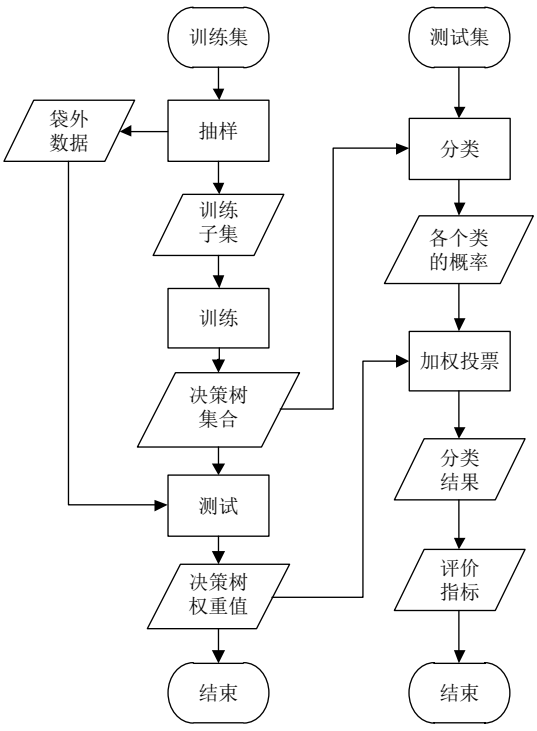

图 2 改进后的随机森林算法流程

3.2 随机森林算法超参数优化

随机森林中超参数的取值对于算法性能有很大的影响, 通常根据经验设置, 但是不同分类问题的最优参数具有较大差异, 经验取值不能取得良好的结果. 若通过在不同的参数之间进行尝试得到最优值需要大量的人力和时间. 对于数据量和超参数范围较大的随机森林应用场景, 经验取值和逐一人工尝试的方法更加不可取. 此外, 随机森林中需要设置的参数较多, 需要得到最优的参数组合. 本文对于数据量和超参数范围较大的随机森林应用场景, 采用随机搜索和网格搜索结合的算法对超参数进行优化, 能够高效的得到超参数的最优值.

网格搜索算法是在所有候选的参数选择中, 通过循环遍历, 尝试每一种可能性, 选择出表现最好的参数作为最终的结果. 由于要尝试每一种可能, 这种算法在数据量、超参数数量和范围较大的情况下需要很长的时间, 性能较差. 随机搜索算法区别于网格搜索的暴力搜索方式, 利用随机数去求函数近似的最优解. 这种算法找到近似最优解的效率高于网格搜索[16], 但可能出现局部最优解的情况. 本文提出随机搜索和网格搜索结合的超参数调优算法, 首先使用随机搜索进行粗选缩小超参数范围, 然后使用网格搜索得到超参数的最优值.

本文选择随机森林算法中较为重要的两个超参数: 决策树棵数 k 和候选特征数 m 进行调节优化, 使用交叉验证的平均预测准确率作为算法性能的评价指标. 超参数优化算法的流程如图 3 所示.

具体步骤如下:

(1) 确定决策树棵数 k 和候选特征数量 m 的范围.

(2) 进行随机搜索, 以模型预测准确率作为算法性能的评价指标, 得到搜索结果.

(3) 对搜索结果进行分析, 考虑性能最好的五组超参数取值. 如果五组结果的超参数取值相近或者性能差距较大, 以最优值附近为范围进行一次网格搜索. 如果五组结果的超参数取值差距较大且性能差距较小, 则在多个小范围内进行网格搜索.

(4) 得到最终的超参数取值.

4 实验结果与分析

4.1 实验数据

本文使用 Python 进行算法实现, 借助 scikit-learn 工具包构建代码[17]. 实验环境: 操作系统为 64 位的 Windows 10 系统, 处理器为 Intel Core i5, 内存为 8

G, CPU 为 2.60 GHz, 开发工具为 PyCharm Community Edition 2017.3.3.

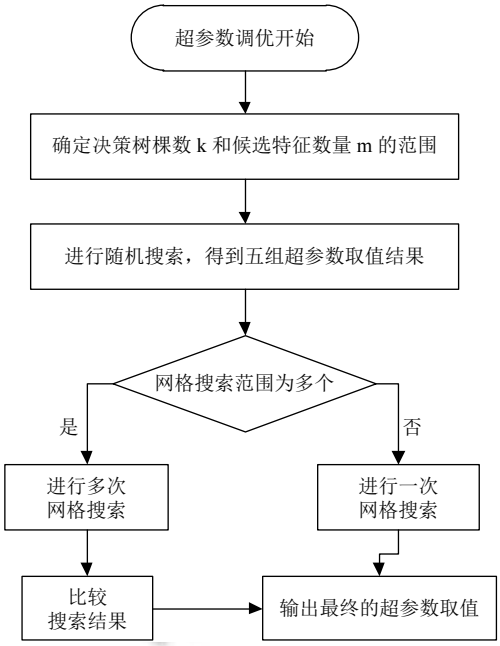

图 3 超参数优化算法流程

20 Newsgroups 数据集是用于文本分类、文本挖据和信息检索研究的国际标准数据集之一. 该数据集有约 20 000 个新闻文档, 包括 20 个类, 每个类分为训练集和测试集两部分, 数据集的文档分布情况如图 4 所示.

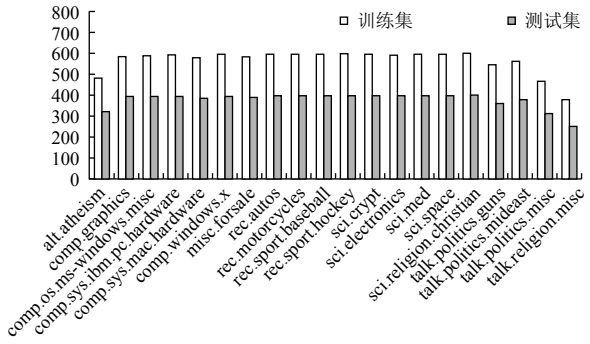

图 4 20 Newsgroups 数据集文档分布情况

4.2 文本分类实验及结果分析

分类前需要对文本语料进行处理, 将其转换为计算机可以处理的数据类型, 本文采用向量空间模型的文本表示方法. 由于数据集是新闻数据, 有大量与类别相关的信息, 分类器不需要从文本中挖掘识别主题就可以获得很好的性能, 所以需要去除这些信息, 然后进行词频统计并使用 TF-IDF 技术处理将文本进行向量化表示. 因为文本中词语较多, 上述处理中使用全部文档中的每一个词作为一个特征形成了高维矩阵, 需要较长的训练和处理时间, 所以使用具有较高性能的 χ^2 统计方法[18]进行特征选取, 最终得到用于实验的数据集.

实验分为两方面, 一方面是改进投票方法的随机森林算法效果分析实验, 对比传统的随机森林算法和改进随机森林算法的性能. 另一方面是随机森林的超参数优化, 证明文中超参数优化算法的可行性同时确定超参数的取值, 比较优化前后算法的性能.

4.2.1 改进随机森林算法效果分析实验

使用传统随机森林算法和改进算法进行文本分类实验, 实验数据为数据集中的所有类别. 分类性能的评价指标为时间和模型预测准确率. 其中模型预测准确率为测试样本中分类正确的样本数占总样本数的比例. 为消除随机性的影响, 分别在决策树棵数为 10、30、50、100、200、300、400 时进行对比实验, 并且每组实验结果都取 10 次运行结果的平均值.

实验结果如表 1 所示. 可以看出, 在不同决策树棵数情况下改进算法在时间上与传统随机森林算法差异不大, 但是模型预测准确率有了一定程度的提高. 本文提出的基于加权投票的改进随机森林算法比传统随机森林算法具有更高的性能.

表 1 改进随机森林算法实验结果

决策树棵树	训练时间 (s)		预测准确率	
	原始随机森林算法	改进随机森林算法	原始随机森林算法	改进随机森林算法
10	3.363	3.552	0.5306	0.5313
30	9.904	10.100	0.5853	0.5892
50	16.698	17.020	0.5943	0.5991
100	33.528	35.017	0.6070	0.6123
200	67.243	68.114	0.6133	0.6179
300	100.436	104.368	0.6146	0.6193
400	134.243	136.865	0.6202	0.6256

4.2.2 随机森林的超参数优化实验

对随机森林算法进行超参数调优, 实验数据为 20 Newsgroups 数据集中的 rec.autos、rec.motorcycles、rec.sport.baseball、rec.sport.hockey、sci.crypt、sci.electronics、sci.med 和 sci.space 八个类. 为了减少随机性对实验结果的影响, 使用交叉验证的规则, 并取测试集上得分的平均值作为评估指标, 得分越高效果

越好. 在随机搜索时设置决策树 k 的范围为 $1<k\leq600$, 候选特征数量 m 的范围为 $1<m\leq146$. 图 5 为随机搜索的实验结果, 根据结果得到最好的五组实验超参数取值如表 2 所示.

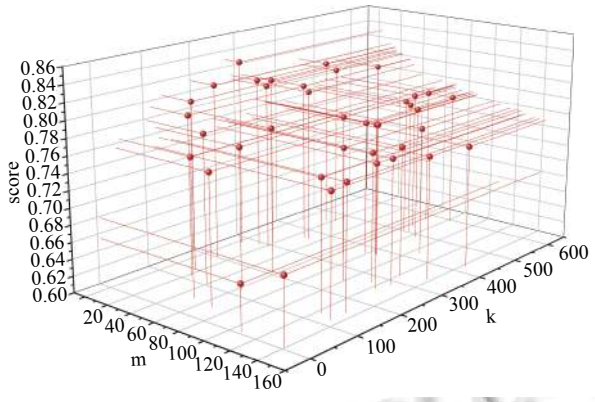

图 5 随机搜索结果

表 2 最好的五组实验

实验序号	k	m	score
1	0.829273	3	279
2	0.812 894	11	201
3	0.809 114	15	459
4	0.808 064	14	291
5	0.805 964	17	316

可以看出, 第一组实验结果明显优于其他四组, 按照前文提出的算法, 只需要在 k=279, m=3 附近进行一次网格搜索. 设置网格搜索的超参数范围, k 的范围为 $250\leq k\leq340$, 步长为 10, m 的范围为 $2\leq m\leq6$, 步长为 1. 图 6 为网格搜索的结果, 可以得出最优参数取值, k=270, m=2, 此时 score 值最高, 为 0.7898.

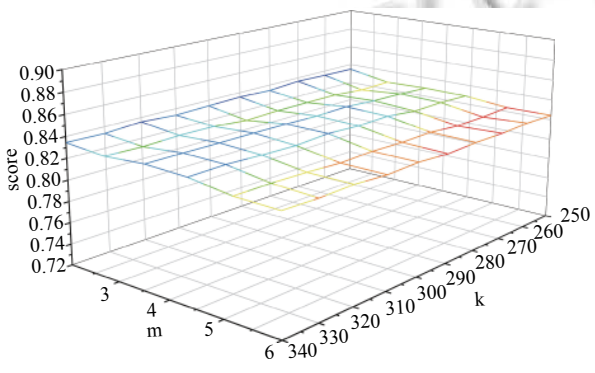

图 6 网格搜索结果

保持其他变量相同, 超参数 k、m 分别为默认值和优化值进行对比, 实验结果见表 3, 可以看出优化后的超参数使得随机森林算法分类能力大大提高.

表 3 超参数调优前后实验结果对比

	k	m	score
默认值	10	91	0.7043
调优	270	2	0.7898

5 结论与展望

针对随机森林分类算法的不足以及文本分类问题中数据量和超参数范围较大等问题, 提出算法改进方案和用于文本分类的分类模型. 一方面使用结合决策树分类效果和类概率的加权投票机制代替传统的投票机制, 提高分类算法性能, 另一方面提出随机搜索和网格搜索结合的算法, 简单高效地实现超参数优化调节. 本文实验在平衡数据集上进行, 具有良好的性能, 在不平衡数据上是否有效需要进一步的研究, 此外为了更好地提高处理大量数据的效率, 今后研究可以考虑结合并行化方法.

参考文献

1 Guo SQ, Gao C, Yao J, *et al*. An intrusion detection model based on improved random forests algorithm. Journal of Software, 2005, 16(8): 1490–1498. [doi: 10.1360/jos161490]

2 卢晓勇, 陈木生. 基于随机森林和欠采样集成的垃圾网页检测. 计算机应用, 2016, 36(3): 731–734.

3 郑志伟, 邱佳玲, 阳庆玲, 等. 随机森林对文本情感分析的应用与 R 软件实现. 现代预防医学, 2018, 45(8): 1345–1348, 1353.

4 张世辉, 刘建新, 孔令富. 基于深度图像利用随机森林实现遮挡检测. 光学学报, 2014, 34(9): 0915003.

5 詹国旗, 杨国东, 王凤艳, 等. 基于特征空间优化的随机森林算法在 GF-2 影像湿地分类中的研究. 地球信息科学学报, 2018, 20(10): 1520–1528. [doi: 10.12082/dqxxkx.2018.180119]

6 杨宏宇, 徐晋. 基于改进随机森林算法的 Android 恶意软件检测. 通信学报, 2017, 38(4): 8–16. [doi: 10.11959/j.issn.1000-436x.2017073]

7 Abellán J, Mantas CJ, Castellano JG. A random forest approach using imprecise probabilities. Knowledge-Based Systems, 2017, 134: 72–84. [doi: 10.1016/j.knosys.2017.07.019]

8 Paul A, Mukherjee DP, Das P, *et al*. Improved random forest for classification. IEEE Transactions on Image Processing, 2018, 27(8): 4012–4024. [doi: 10.1109/TIP.2018.2834830]

9 何珑. 基于随机森林的产品垃圾评论识别. 中文信息学报,

2015, 29(3): 150–154, 161. [doi: 10.3969/j.issn.1003-0077.2015.03.020]
10 贺捷. 随机森林在文本分类中的应用[硕士学位论文]. 广州: 华南理工大学, 2015.
11 田宝明, 戴新宇, 陈家骏. 一种基于随机森林的多视角文本分类方法. 中文信息学报, 2009, 23(4): 48–54. [doi: 10.3969/j.issn.1003-0077.2009.04.008]
12 Breiman L. Random forests. Machine Learning, 2001, 45(1): 5–32. [doi: 10.1023/A:1010933404324]
13 Cutler DR, Edwards Jr TC, Beard KH, *et al*. Random forests for classification in ecology. Ecology, 2007, 88(11): 2783–2792. [doi: 10.1890/07-0539.1]
14 李毓, 张春霞. 基于 out-of-bag 样本的随机森林算法的超参数估计. 系统工程学报, 2011, 26(4): 566–572.
15 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. 软件学报, 2006, 17(9): 1848–1859.
16 Bergstra J, Bengio Y. Random search for hyper-parameter optimization. Journal of Machine Learning Research, 2012, 13(1): 281–305.
17 Bowles M. Machine Learning in Python®: Essential Techniques for Predictive Analysis. Indianapolis: John Wiley & Sons, 2015.
18 Rogati M, Yang YM. High-performing feature selection for text classification. Proceedings of the 11th International Conference on Information and Knowledge Management. McLean, VA, USA. 2002. 659–661.