文章编号: 2096-1618(2019)03-0274-04

一种针对不均衡数据集的SVM决策树算法

黄 勇[1], 魏 乐[2,3]

(1.成都信息工程大学计算机学院,四川 成都 610225; 2.成都信息工程大学软件工程学院,四川 成都 610225; 3.软件自动生成与智能服务四川省重点实验室,四川 成都 610225)

摘要: 针对文本分类问题中常遇到的数据分布不均的情况,提出一种新的SVM决策树算法。算法在构造分类器结点时,运用动态规划的思想,寻找类别数和样本数量同时最优的分配方案。实验结果表明,该方法比基于完全二叉树的SVM分类器准确率有明显提升。

关 键 词: 大数据; 自然语言处理; 动态规划; 完全二叉树; 支持向量机; 文本分类; 机器学习

中图分类号: TP391 文献标志码: A

doi: 10.16836/j.cnki.jcuit.2019.03.012

0 引言

类别间的语料分布不均衡是降低许多分类算法准确率的重要因素,许多分类器在分类时都倾向于将语料分为大类,所以就造成分类算法准确率降低[1]。然而不均衡的分类问题在现实生活中却是普遍存在的,并且多数时候分类中的小类才是需要关注的地方。比如网络攻击日志、非法信用卡交易信息等属于小类,通常分类器分类时小类语料准确率低,进一步增加了发现非法的记录难度[2]。

当前主要有以下6种解决方案:

(1) 重新采样。欠采样,通过减少大样本个数来平衡数据集,当数据量比较多时考虑使用该方法; 当数据量不够时考虑使用过采样,与欠采样相反,该方法是通过增加小样本数量来平衡数据集。常用的方法有重复选取、根据分布函数自主生成或合成数据等过采样等方法来增加小样本数量。

(2) 组合不同的重采样数据集[3]。建立n个模型,每个模型使用小类别的所有样本和大类别的n个不同样本。假如想合并10个模型,那么保留比如1000条小类别,再随机抽取10000条大类别,将10000条样本分成10块,并训练10个不同的模型。

(3) 转化为一分类问题。如果二分类问题正负样本比例极其不平衡,可以把问题转化为一分类问题或异常检测问题。这些方法的重点不是捉类间的差别,而是为其中一类进行建模,包括经典的One-class SVM等。

(4) 多模型Bagging。上面的方法全是从数据集角度解决不均衡的,虽然可以产生合适的样本集,但是往往不能保证鲁棒性。从模型方面下手,可以考虑使用集成学习的多模型Bagging。Bagging算法的特点是采用有放回的随机采样,然后将多个弱学习器组成一个强学习器。

(5) XGBoost。Boosting中的XGBoost算法在设计上较好地考虑了不均衡数据集的情况,并且目前的XGBoost接口中有专门的参数用来针对不平衡数据[4]。

(6) 先使用无监督的聚类,再使用有监督的分类。先对大样本进行聚类,筛选出数量接近小样本数量的若干个数据集合,然后使用样本数相等的若干个正负样本进行有监督的分类。

上面这些处理方法存在一些问题。比如,直接复制法在数量上扩充了数据集,但是仅仅是在小类别语料已有的空间内扩充,并没有往空间外扩张,面对大类别时分割的超平面依然倾向于大类别; 插值法没有考虑本类别独有的特点,扩充存在很大盲目性; 欠抽样法通过丢弃数据达到平衡,其实浪费了大量有价值数据; 算法层面设置权重需要考虑设置多大权重合适,以及和不同类别做分类时该怎么调整权重,通过设置权重具有一定的盲目性[5]。

提出的算法在不改变原数据集的情况下,重新规划决策树每个结点上的数据分配,使分类器训练数据分布更加均匀,分割超平面不再偏向于样本量多的类别,进而提升分类器的准确率。

1 完全二叉树决策树

在具有相同叶子结点数的情况,完全二叉树是高度最低的二叉树,将完全二叉树和SVM结合,既可以

收稿日期: 2019-02-24
基金项目: 四川省科技计划重点研发项目资助(2017GZ0309); 四川省教育厅青年基金重点资助项目(16ZA0208)

快速达到叶子结点又可以将只能二分类的 SVM 实现多分类。

对于给定的数据集 $D=\{d1,d2,\cdots,dn\}$,其中 $di=<di1,di2,\ldots,dih>$表示 D 中的第 i 个样本($i=1,2,\cdots,n$),数据集的属性集为$\{L1,L2,\ldots,Lh\}$,所以 dij 表示的是第 i 个样本的第 j 个属性($j=1,2,\cdots,h$)。同时给出类标号的集合 $C=\{C1,C2,\cdots,Cm\}$。对于给定数据集 D,决策树可以定义为具有如下 3 个性质的树:

(1) 每个非叶子结点指定一个分裂属性 Li。

(2) 每个分支指定一个关系谓词,这个关系谓词用来指明分裂的父节点。

(3) 每个叶子结点指定一个类符号 Cj。

处理步骤为:

(1) 将所有类别重新组合划分为两个新类,同时将训练数据也划分为对应的两组,于是问题被转化为 SVM 二分类问题。

(2) 将每个包含类别个数大于 1 的新类再分成两个新类,同样将训练数据划分为对应的两组训练 SVM 分类器。

(3) 重复步骤(2),直到每个新类只包含一个类别为止,这个叶子结点就是分出的类别。

进行预测时,当分类进行到叶子结点为止,叶子结点对应的类就是分出的类别[6]。

如图 1 所示,二叉树的结构对分类器效果有很大影响,完全二叉树树高最低的特点可以减少分类次数,进而减少训练和预测的时间。并且由于完全二叉决策树叶子结点即类别,没有出度为 1 的结点,这样就进一步减少了分类器个数[7]。

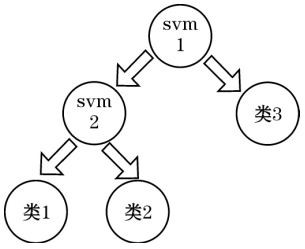


图 1　SVM 完全二叉树

2　针对不均衡数据集的 SVM 决策树算法

算法针对数据集不均衡的特点,在构造分类器结点上做出改进,使每个分类器的训练数据分布更加均匀。

假设一个无序的元素个数为 $2N$ 的正整数数组,要把它分割为元素个数为 N 的两个数组(如果是 $2N+1$ 个元素,则分割成 $N+1$ 和 N,本节只以 $2N$ 为例),并且要求两个子数组的元素和是最接近的。假设有数组 $A[1,\cdots,2N]$,且所有元素的和是 SUM,修改动态规划解 0-1 背包问题的策略,令 $S(k,i)$ 表示前 k 个元素中任意 i 个元素的和的集合[8]。则有

$$S(k,1)=\{A[i]\mid 1\leqslant i\leqslant k\}$$

$$S(k,k)=\{A[1]+A[2]+\cdots+A[k]\}$$

$$S(k,i)=S(k-1,i)\ U\{A[k]+x\mid x\in S(k-1,i-1)\}$$

依据递推公式,找出元素个数为 N 元素之和最接近 SUM/2 的组合,与 SUM/2 最接近的和就是与另一个子数组和差值最小的数组,便是要求的数组,时间复杂度为 $O(2^N)$[9]。

在计算过程中只需找出和不大于 SUM/2 的数组。数组规划过程中总值相同的组合不用逐一考虑,另外当总值大于 SUM/2 以后的组合也不用考虑。所以不需要计算出所有的 $S(2N,N)$,只需要从总值为 SUM/2 到 1 分析一遍,只需要给每个集合设置一个标志数组,查找 [1,SUM/2] 中哪些值是可以计算出来的,其中最大的值就是要求的答案[10]。

在完全二叉树分类器结点的构造上,使用数组规划方法划分新的类别,使类别个数和样本个数同时达到最优,即针对不均衡数据集的 SVM 决策树算法[11-12]。

3　实验及结果分析

实验使用的数据集有以下 3 个: 公开数据集 sklearn 的英文文本数据集 The 20 newsgroups text dataset,共 18846 条,划分为 20 类; THUCNews 是由新浪新闻的历史数据经过过滤筛选得到的,有 74 万篇新闻被分为 14 类,现均匀选取 32200 条; 今日头条中文新闻(文本)分类数据集,原数据集有 382688 条,被分为 15 类,现均匀选取 30000 条。数据集特点如表 1 所示。

表 1　数据集特征

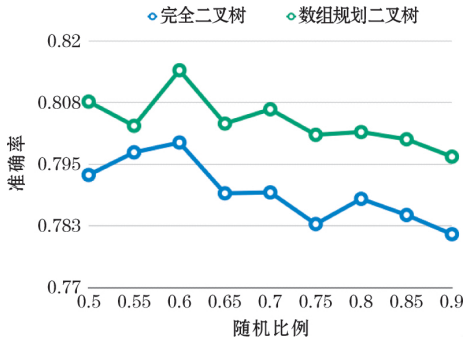
datasets	Classes	Samples total	Features
20 newsgroups	20	18846	text
THUCNews	14	32200	text
Toutiao_dataset	15	30000	text

原数据集分布比较均匀,为了构造不均衡的数据分布,在实验中对每个类别数据量进行处理,处理的原则是保留一部分数据作为基础数据,另一部分数据随机选择。

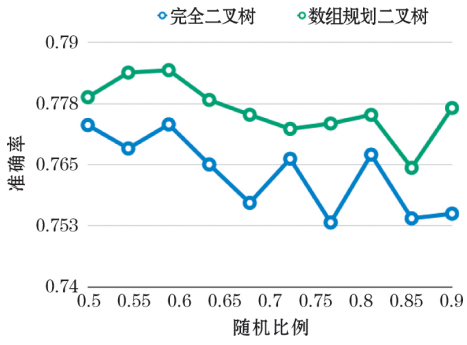
实验环境是 Google Colab,CPU 是 Intel Xeon

2.20 GHz,RAM 是12.7 Gb,实验中使用的随机数的 seed 是 10001。

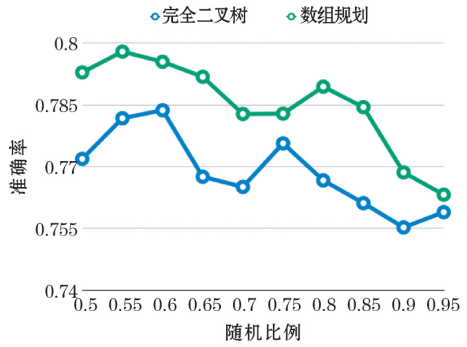
实验中逐步调整随机选取部分占比,对比两种决策树的准确率,进而验证在处理不均衡数据时,提出的方法是否整体上比完全二叉树决策树准确率更高。不均衡数据的占比设置为从 50%到 90%变动,每 5%取一个抽样进行对比。



(a) 20 newsgroups 数据集



(b) THUCNews 数据集



(c) Toutiao_dataset 数据集

图 2 3 种数据集的准确率折线图

实验结果对比图如图 2 所示,从图中可以看到,随着随机比例的增加,也就是随着不均衡程度增加,准确率整体呈下降趋势。在准确率下降过程中,数组规划二叉树的准确率比完全二叉树高1.5%左右。结果说明,在处理不均衡数据集时,经过数组规划的决策树明显比具有相同树形结构的完全二叉树决策树的准确率更高。

4 结束语

基于完全二叉树的 SVM 决策树在处理多分类问题时具有识别速度快、识别准确率高等优点,因此树型的分类算法的应用比较广泛。针对文本数据不均衡的特点,提出了一种基于完全二叉树树形结构的经过数组规划处理的 SVM 决策树。经过一系列的实验对比,证明了新方法在处理不均衡数据集上确实比一般的完全二叉树决策树准确率更高。下一步计划在做数组规划时考虑更多因素,如两个类别的误分概率等,进而进一步提升准确率。

参考文献:

[1] 陶新民,郝思媛,张冬雪,等.不均衡数据分类算法的综述 [J].重庆邮电大学学报(自然科学版),2013,25(1):101-110.

[2] 李诒靖,郭海湘,李亚楠,等.一种基于 Boosting 的集成学习算法在不均衡数据中的分类 [J].系统工程理论与实践,2016,36(1):189-199.

[3] 孙晓燕,张化祥,计华.用于不均衡数据集分类的 KNN 算法 [J].计算机工程与应用,2011,47(28):143-145.

[4] 杜红乐,张燕.密度不均衡数据分类算法 [J].西华大学学报(自然科学版),2015,34(5):16-23.

[5] 崔建,李强,刘勇,等.基于决策树的快速 SVM 分类方法 [J].系统工程与电子技术,2011,33(11):2558-2563.

[6] Segata N,Blanzieri E.Fast and Scalable Local Kernel Machines [M].JMLR.org,2009.

[7] Dorff K C,Chambwe N,Srdanovic M,et al.BD-Val: reproducible large-scale predictive model development and validation in high-throughput datasets [J].Bioinformatics,2010,26(19):2472-2473.

[8] 程凤伟.一种基于决策树的 SVM 算法 [J].太原学院学报(自然科学版),2017(1):33-36.

[9] 王琛,王云,陈丽芳,等.哈夫曼树 SVM 在空气质量等级分类中的应用 [J].智能计算机与应用,2016,6(1)64-67.

[10] 陈丽芳,陈亮,刘保相.基于粒计算的哈夫曼树 SVM 多分类模型研究 [J].计算机科学,2016,43(1):64-68.

[11] 孙怀影,耿寅融,单谦.求解 0-1 背包问题的一种新混合算法[J].计算机工程与应用,2012,48(4):50-53.

[12] Mensch A, Blondel M. Differentiable Dynamic Programming for Structured Prediction and Attention[J].2018.

An SVM Decision Tree Algorithm for Unbalanced Data Sets

HUANG Yong[1], WEI Le[2,3]

(1.College of Computer Science, Chengdu University of Information Technology, Chengdu 610225, China; 2.College of Software Engineering, Chengdu University of Information Technology, Chengdu 610225, China; 3.Sichuan Key Laboratory of Software Automatic Generation and Intelligent Services, Chengdu 610225, China)

Abstract: A new SVM decision tree algorithm is proposed for the uneven distribution of data commonly encountered in text classification problems. When constructing the nodes of classifier, the algorithm uses the idea of dynamic programming to find the optimal allocation scheme of both the number of categories and the number of samples.The experimental results show that the proposed method has a significantly better accuracy than the SVM classifier based on the complete binary tree.

Keywords: big data; natural language processing; dynamic programming; complete binary tree; support vector machine; text classification; machine learning