

基于类重叠度区分的大规模云平台 任务终止状态预测

代丽萍¹,王敬雄¹,李为丽¹,刘春红^{1,2*},程渤³

(1. 河南师范大学计算机与信息工程学院, 河南 新乡 453007; 2. 智慧商务与物联网技术河南省工程
实验室 河南 新乡 453007; 3. 北京邮电大学网络与交换技术国家重点实验室 北京 100876)

摘要:大规模云平台任务终止状态的预测是云资源调度策略优化的关键步骤。本文以 Google 云平台的计算调度系统 Borg 为对象进行研究,针对任务的各种终止状态极度不均衡和类重叠的问题,提出了一种类重叠度区分的自定义步长-梯度提升决策树 (SP-GBDT) 任务终止状态预测方法,对任务终止状态进行细粒度的多分类预测,提高少数类任务状态的预测准确率。首先将终止状态的多个类别拆分成若干个二类组合,通过支持向量数据描述模型 (SVDD) 筛选出类重叠度较低的最优二类组合。然后,分别对最优的二类组合进行扩展采样比例的自定义步长欠采样。最后构建梯度提升决策树模型,将欠采样之后的数据进行多分类。在 Google 云平台的运行监控日志数据集上进行验证,通过对比预测结果和预测过程的可解释性分析,SP-GBDT 模型能够很好地降低数据集的不均衡比例以及类重叠的程度。与决策树和随机森林等常用多分类预测方法相比,所提算法的 F1-score 分别提高了 30.39% 和 18.26%。

关键词:终止状态;不均衡多分类;类重叠度;欠采样;可解释性

中图分类号:TP312 文献标识码:A 文章编号:1673-4793(2021)02-0044-10

Task-termination-statuses prediction method based on discrimination of class overlap in large scale cloud

DAI Liping¹, WANG Jingxiong¹, LI Weili¹, LIU Chunhong^{1,2}, CHENG Bo³

(1. College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China; 2. Engineering Lab of Intelligence Business & Internet of Things, Xinxiang 453007, China; 3. State Key laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: Task termination statuses prediction is one of the key technologies to realize resource scheduling optimization in large-scale cloud platform. In this paper, using the Google Cloud Platform's computing scheduling system Borg as the object, aiming at the problem that various termination statuses of tasks are extremely unbalanced and overlapping, a Self-Paced λ -Gradient Boost Decision Tree (SP-GBDT) is proposed to predict task termination statuses. Fine-grained multi-classification prediction of task termination state is carried out to improve the accuracy of predicting a few classes of task states. First, the termination state is divided into several two-class combinations, and the best two-class combination with low class overlap is selected by Support Vector Data Description model (SVDD). Then, the optimal combination of the two classes is under-sampled for the custom step of the extended sampling scale. Finally, a gradient lifting decision tree model is constructed to classify the under-sampled data. Verification on the operation monitoring log data set of the Google cloud platform, the experimental results show that the SP-GBDT model can reduce the imbalance ratio of the dataset and the degree of

class overlap through the study of the interpretability of the prediction process and results. Compared with the commonly used multi-classification prediction methods such as decision trees and random forests, the F1-score of the proposed algorithm increased by 30.39% and 18.26%, respectively.

Key words: termination statuses; unbalanced multi-classification; class-overlap; under-sampling; interpretability

1 引言

大规模云平台运行着成千上万项应用,云集群管理系统调度各种资源,以保障应用的正常运行,并提高云资源的利用率^[1]。目前,云资源调度策略的优化已成为提高资源利用率的重要手段。其中,基于机器学习的任务终止状态预测结果,作为调度器采取相应调度策略的决策依据,已成为云资源管理优化的研究热门^[2-10]。

在分布式云平台的调度管理中,用户的应用请求以作业的形式提交到数据中心,每个作业由调度器分配计算和存储资源,作业的一次调度运行,称为任务。作业包含一个或多个任务。其中,任务终止状态包含驱逐、失败、完成和被杀^[1]。提前预测任务的终止状态,对不能成功完成的任务采取相应的调度策略,可有效节约资源,缩短任务的运行时间,提高资源的利用率^[6-7,9-10]。

云平台失败任务预测研究分为任务终止状态影响因素的分析和终止状态预测方法两个领域。任务终止状态影响因素的分析从失败预测的可行性、任务和资源请求等属性的相关性以及故障表征研究等进行分析^[3-6]。Jassas等人^[2]探索了失败作业与CPU资源请求等因素之间的相关性。Islam等人^[3]对task进行了故障表征研究,并探索云平台进行失败预测的可行性。基于机器学习技术的终止状态预测是近年来的研究热点。利用云平台的监视日志收集作业/任务的属性和运行状态数据,在任务终止状态影响因素研究的基础上,使用机器学习技术,建立终止状态预测模型。目前常用支持向量机(SVM)^[4],集成模型^[5]和神经网络(NN)^[7]进行预测。P. Padmakumari等人^[5]使用bagging集成学习策略,集成多个弱分类器进行失败预测,并获得较好的预测精度。越早期的失败预测会带来更好的收益,然而,越早期预测提供给预测模型的运行状态信息量越少。Chen等人^[8]选择在任务运行的1/2时刻,使用递归神经网络(RNN)模型预测,但预测时期相对较晚,且精度不高。Liu等人^[9]提出

了利用作业间的结构化信息,弥补早期预测时信息量的不足,在任务执行1/3时刻进行预测的联合预测方法,但该方法将任务终止状态简化为成功和失败两类。

为了更准确地对云平台不同终止状态任务采取不同的调度策略,需要进一步对任务的终止状态预测进行细分,即不仅预测任务是成功还是失败,还需要进一步细分失败是驱逐、失败或被杀的状态,从而采取提前终止或继续运行等不同的调度策略。Rosa等人^[7]提出的嵌套神经网络(Nested Neural Network, Nested Prediction)预测方法,进行任务多种终止状态的预测,并分析了作业终止状态对任务终止状态的影响。但是,该方法预测准确率不高,如果某作业终止状态对应的任务数量很少,则可能导致更低的准确率。

本文以Google云平台的计算调度系统Borg^[1]为研究对象,利用Google公开的Borg代表性工作负载监控日志(由<https://github.com/google/cluster-data>下载),该日志有一个月的监控数据,统计分析其失败和被杀的任务,其中,任务有多个事件在规定时间内,最后一个事件的终止状态被认为是任务的最终终止状态。任务的事件是指提交、调度、运行和终止这四个状态之间的转换,运行过程中各个事件的最终执行概率,统计结果如图1所示。

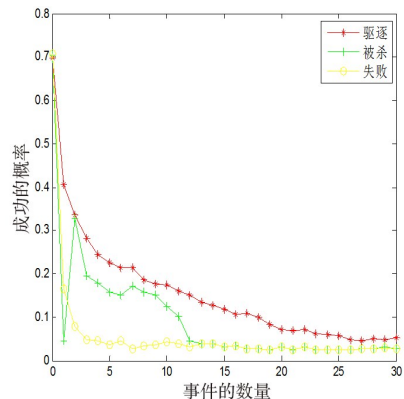


图1 各种事件对应的任务成功执行概率

由图1可见,事件成功执行的概率随着驱逐和失败事件的增加而下降。由于驱逐的任务多是由于任

务前期所请求资源不足或优先级过高导致,任务运行后期可能会被满足,可以再次重新调度。失败的任务是因为任务本身出现问题而终止,所以即使重新调度,失败的概率依然很高,而被杀的任务多是因为被用户自己取消或者所依赖的任务失败,可以再次重新调度,但如果调度后仍不能执行,则最终终止的概率很大。

因此,对于不同任务终止类型需要采取不同的调度策略,可以提高机器运行效率、节约资源。为了对任务态采取更细粒度的调度策略,需要构建多种终止状态的预测模型。

任取3天的监控日志数据,统计分析任务的终止状态。任务终止状态中,驱逐、失败、完成、被杀的比率为1:2:3031:343,其中,最大不均衡比例为1:3031。终止状态为完成和被杀的任务数量远远高于驱逐和失败的任务,任务类别属于极度类不均衡(class imbalance)。当处于极度不均衡的状态时,很难对少数类的任务,即驱逐和失败状态,进行有效的预测。极度不均衡的训练集训练出来的分类器更偏向于多数类,从而使少数类的分类性能下降。

同时,由于Borg系统中同一个作业包含的任务,其大部分属性相同,例如静态特征优先级、调度类、CPU请求资源、RAM请求资源和DISK请求资源等特征属性基本一致。而不同类别的任务往往在这些特征属性上具有相似的数值,导致数据集在极度不均衡的同时,不同的终止状态具有明显的类重叠,四种终止状态类的重叠程度如图2所示。图2将失败预测特征使用PCA降维,选择任务的主要二维特征作为横纵坐标,不同颜色的点区分任务的不同终止状态,图2可见,驱逐和失败状态被成功状态覆盖,成功和被杀状态有大面积的重叠。

由于极度不均衡和类重叠的问题,常用的多分类预测方法不能获得良好的预测性能。针对该问题,本文提出了自定义步长-提升决策树(Self Paced -Gradient Boosting Decision Tree, SP-GBDT)模型,对极度不均衡的原始数据集进行扩展比例的自定义步长欠采样之后,选取重叠度较低的最优二类组合,构建梯度提升决策树模型,将欠采样之后的数据进行多分类。该方法能有效解决数据集的不均衡程度和类重叠程度,筛选出信息含量更为丰富的数据,提高模型对于少数类任务的正确分类能力,获得很好的分类性能。

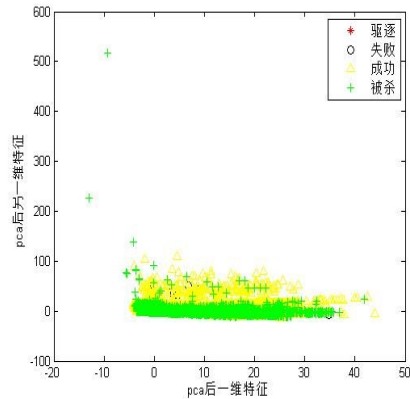


图2 四种终止状态的重叠图

2 基于类重叠度区分的不均衡多分类

为实现任务不同终止状态的准确预测,提出了SP-GBDT预测方法。算法架构图如图3所示。

设待预测数据共有 n 类,为计算各个任务不同终止状态之间的重叠度,以及更好地对二类组的任务数据进行自定义步长的欠采样,将不同任务的多个终止状态分为若干个二类组。首先将不均衡的数据集按7:3的比例,随机分为训练集和测试集。将训练集分为 $n(n-1)/2$ 二类的组合,并根据各个二类组合的平衡比例和类重叠程度,选择最优的二类组合;然后针对数据集极度不均衡的情况,扩展欠采样的比例,以获取更多的信息含量丰富的任务,同时使得多类和少类的数据具有较为平衡的比例;最后使用欠采样后的新训练集训练模型,使用原始测试集进行多分类预测。

2.1 类重叠度区分

数据集 X 包含 n 个任务,其中 $X = \{x_1, x_2, \dots, x_n\}$ 。任务 x_i 包含了静态和动态特征, $i = \{1, 2, \dots, n\}$, $n \in N^+$ 。标签 Y 包含 n 个任务的终止状态,即 $Y = \{y_1, y_2, \dots, y_n\}$ 。任务的终止状态 y_i 由一维类别标签组成,其中 $y_i \in \{2, 3, 4, 5\}$,2表示任务终止状态为驱逐,3表示任务执行失败,4表示任务执行成功,5表示任务在执行过程中被杀。

任务各个终止状态的重叠程度对其可分离性是有一定影响的。至少有两个类 C_1, C_2 同时出现在区域A的概率大于0,即对任意的重叠任务数据 x_i , $p(x_i^j|C_1) > 0$,且 $p(x_i^j|C_2) > 0$ 。如果重叠区域较大,则

分类器的性能指标便会降低。

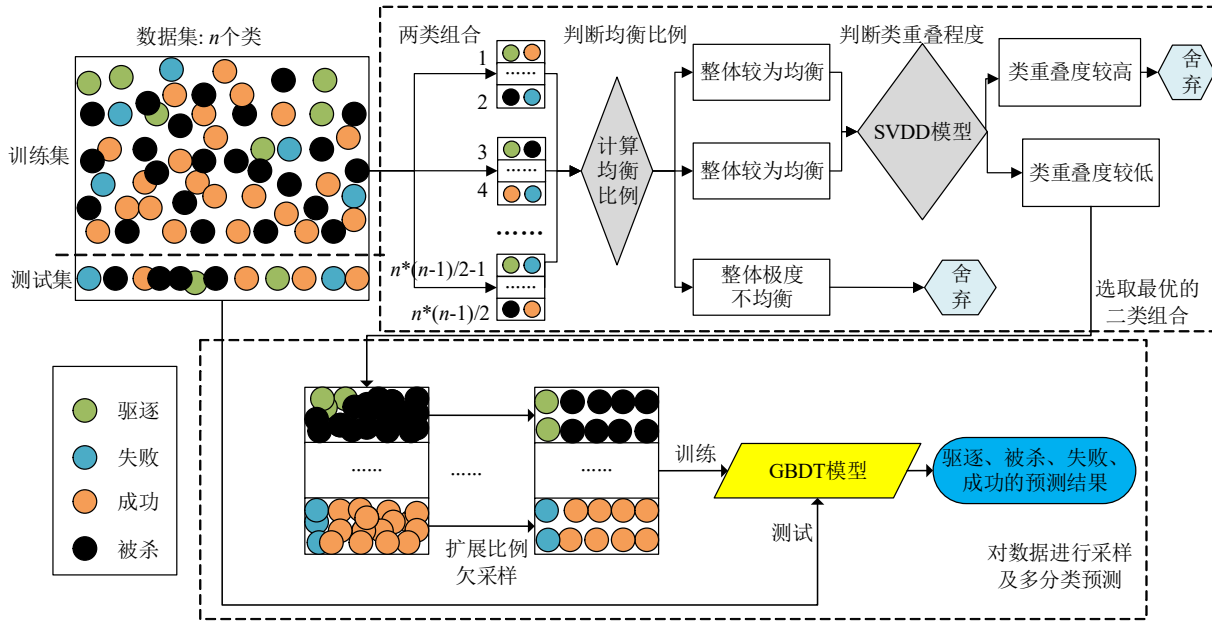


图3 SP-GBDT架构图

为降低不同类别的重叠度,利用支持向量数据描述算法(Support Vector Data Description, SVDD)^[11]方法识别互相重叠的任务数量,从而确定重叠区域。然后选取均衡比例较好且重叠度最低的二类组合为最优的组合。

SVDD模型的决策函数如公式(1)所示:

$$f(x) = \text{sgn} \left[R^2 - \sum_{ij=1}^n \alpha_i \alpha_j K(x_i, x_j) \right] + 2 \sum_{i=1}^n \alpha_i K(x_i, x) - K(x, x) \quad (1)$$

其中 $\text{sgn}(\cdot)$ 是阶跃函数,取值在-1、0、1之中。

将任务数据输入到SVDD模型中,其决策函数的值若满足以下条件,则可对任务是否为重叠样本进行判定。其中,SVDD算法参数默认选择为0.5。

$$\begin{cases} f(x) > 0 & \text{内点} \\ f(x) = 0 & \text{支持向量} \\ f(x) < 0 & \text{外点} \end{cases} \quad (2)$$

利用SVDD算法对任务数据中的每个终止状态类别进行单类学习,基于各个单类模型的决策函数 $f(x)$ 进行计算比较。若任务样本数据在至多一个单类模型上的决策函数值大于或等于0,则它处于非重叠区域;若任务在至少两个单类模型上的决策函数值大于或等于0,则它处于重叠区域。

以计算任务终止状态为完成和失败的两类样本

的重叠度计算为例说明:第一步,完成样本和失败样本分别训练了SVDD的模型为 M_1 和 M_2 。第二步,合并完成和失败两类样本,将合并后的数据分别输入 M_1 、 M_2 模型中,得到决策函数的值。第三步,根据一个样本中至少两个决策函数值大于等于0即为重叠区域,计算完成和失败两类中重叠样本个数。第四步,重叠样本个数比上合并样本总数即为完成和失败两类的重叠比例。

2.2 SP-GBDT

SP-GBDT使用SP欠采样(Self-paced under-sampling)^[12]减少带有各类比例不均衡的任务数据对模型的影响。采用GB算法和决策树(CART)作为基学习器,其中,决策树能最大化拟合训练数据。将GB算法和CART决策树模型结合,得到GBDT模型。GBDT模型能通过梯度提升的方法集成多个决策树,损失函数是deviance,且将负梯度作为上一轮单颗决策树犯错的衡量指标,在下一轮学习中通过拟合负梯度来纠正上一轮犯的错误。因此,SP-GBDT模型解决极度不均衡的同时,增加边界样本的重要性。

SPE欠采样中硬度函数和自定义步长的定义如下:

(1)硬度函数:硬度函数的定义如公式(3),每个任务样本具有一个硬度值,将多类的任务数据按照硬度值分成 k 个箱子,其中 k 是超参。每个箱子都有一

个特定的硬度级别,通过保持每个箱子中的总硬度值相同,对数据集进行欠采样,使其达到一个平衡的程度。硬度函数如公式(3)所示:

$$H(x,y,f) = |f(x) - y| \quad (3)$$

其中, $f(x)$ 为GBDT模型的目标函数,即硬度函数的值和使用的模型相关。

然而在训练过程中,只通过协调箱子的硬度值,使分类器逐步对训练集进行拟合,会导致信息含量并不丰富的任务数据样本的数量不断增加。而且这些样本由于信息量较小,大大降低了迭代的学习过程。因此,本文引入“自定义步长因子(Self-paced因子)”来自动协调上述不足。

(2)自定义步长因子:Self-paced因子,记为 η ,能协调各个箱子的硬度值,且能逐步调控箱子的抽样概率。当 η 变大的时候,模型更加关注分类较为困难的样本,而不是只简单的协调硬度值。

首先,将多类任务样本分为多个二类组合,根据其不平衡比例和类重叠度选出最优的二类组合;然后分别对二类组合进行SP欠采样;最后进行多分类预测。具体步骤如下所示:

Step 1:将二类组合中任务样本的数据集分为训练集和测试集,根据训练集中任务不同终止状态的数量,分为多类和少类。

Step 2:第 $i = 1$ 次迭代,其中 $i = 1, 2, \dots, M$

1)初始化GBDT集成模型 $f_0(x)$ 。

2)根据硬度函数 $H(x,y,f)$ 和 $f_0(x)$,将训练集中的多类切分成 k 个箱子,分别是 B_1, B_2, \dots, B_k 。

3)第 j 个箱子平均硬度值为

$$h_j = \sum_{s \in B_j} H(x_s, y_s, f_i) / |B_j|, \forall j = 1, 2, \dots, k。$$

4)更新自定义因子 $\eta = \tan\left(\frac{i\pi}{2\pi}\right)$ 。

5)第 j 个箱子的非标准化采样权重 $p_j = \frac{1}{h_j + \eta}$ 。

6)对含有 $\sum_{m=1}^M p_m |P|$ 个样本的第 j 个箱子进行下

采样。

7)使用最新的欠采样数据子集对GBDT模型 $f_i(x)$ 进行更新训练。

Step 3:在 $f_M(x) = \sum_{m=1}^M \sum_{j=1}^J \varepsilon_{jm} I$, $x \in R_{jm}$ 时,得到最

终的集成模型,将测试集输入到训练好的集成模型

中,得到多分类的结果。

2.3 算法描述

SP-GBDT模型针对云平台任务终止状态预测时,不同终止状态的极度不均衡和类重叠的问题,首先根据均衡比例和重叠度,得出训练数据集的最优二类组合,然后构建模型,将二分类欠采样扩展为多分类欠采样。算法描述如算法1所示。

算法1:基于类重叠度区分的任务终止状态预测算法

输入:任务的数据集 X

输出:任务的终止状态

1. 初始化参数:数据集类别数 n ,大类数 $t, t = \lceil n \times (n - 1) / 2 \rceil$,其中 $\lceil \cdot \rceil$ 为向上取整函数。
2. 将数据集按比例随机分为训练集和测试集。
3. 训练集种 n 类可分类为 $n \times (n - 1) / 2$ 种二类, t 个二类组合,其中二类组合为包含所有 n 类样本的组合
4. $i = 1, 2, \dots, t$
5. 计算第 i 大类包含的各二类的均衡比例。
6. IF 第 i 大类总体比例不均衡
7. 则该方法不为最优
8. else 计算第 i 大类中各个二类的重叠度之和。
9. END
10. 得到重叠度集合 $O = \{o_1, o_2, \dots, o_t\}$,从中选取最小值,其对应的二类组合则为最优二类组合,即选取总体比例均衡且重叠率较低的二类组合。
11. 将最优的二类组合输入SP-GBDT模型中,得到多分类结果。

3 实验结果与分析

采用大型通用云平台Google的监控日志数据集进行验证。筛选了Google集群日志前三天的数据,该样本数据集中共有10473个job,包含1665280个任务。

验证实验分为3组:(1)类重叠度的计算和最优欠采样比例的选择;(2)欠采样参数的选择;(3)预测结果的比较和可视化分析。

实验在集群系统上进行,该集群由两个E5-2630 CPU, 32 G RAM和两个Tesla K20m图形卡。在Python 3.6和MATLAB R2016a环境下运行。

3.1 数据预处理及评价指标

由于云平台数据集从真实环境中采集获取,数据集存在部分噪声和缺失值等问题,首先进行数据的清洗处理。清洗过程如文献[6]所示。

接着进行归一化。由于任务各个特征属性的度量单位不同,通过Z-score函数将由任务样本数据标准化。其中Z-score函数的如公式如(4)所示:

$$(x_i)^* = \frac{x_i - \bar{x}_i}{s} \quad (4)$$

其中 s 是特征的标准差, \bar{x}_i 是任务特征的平均值。

多分类评价指标分为宏平均(Macro-average)和微平均(Micro-average),以召回率(Recall)为例,Macro-R表示宏平均的Recall, Micro-R表示微平均的Recall。其中Macro-R为将 n 分类的预测值拆成 n 个二分类的预测值,计算每个二分类的Recall, n 个Recall的平均值即为Macro-R。Micro-R为将 n 分类的预测值拆成 n 个二分类的预测值,将 n 个二分类评价的TP、FP、TN、FN对应相加,计算Recall。针对极度不均衡的多分类来说,Macro-R受样本数量少的类别影响较大,比Micro-R更合理。所以本文在测试数据集上,选择宏平均来度量分类器性能,更侧重对小类判别的有效性评价。

G-mean是评价不均衡多分类模型的常用指标,正例准确率与负例准确率的综合指标。因为数据集极度不均衡时,模型可能会对多类数据产生过拟合现象,导致实验结果的预测精度很高,但此模型对少数类的预测正确率将会很低。

多分类各评价指标的计算公式如(5~9)所示。

$$Macro - R = \frac{1}{n} \sum_{i=1}^n R_i \quad (5)$$

$$Macro - P = \frac{1}{n} \sum_{i=1}^n P_i \quad (6)$$

$$Macro - F1 = 2 \times \frac{Macro - R \times Macro - P}{Macro - R + Macro - P} \quad (7)$$

$$G - mean = \sqrt{Macro - R \times Macro - P} \quad (8)$$

$$AUC = Area Under precision - Recall Curve \quad (9)$$

3.2 类重叠度的计算

将任务样本各个终止状态分为若干种二类组合,使用SVDD模型计算各个二类组合的不均衡比例以及重叠度,以获取不均衡比例较低且重叠度最低的二类组合。具体数值如表1所示。

表1 各种二类组合的不均衡比例及重叠度

组合方式	类别	任务总数	不均衡比例	重叠度	重叠度之和
第一种	驱逐:完成	312277	1:3031	0.1358	0.1069
	失败:被杀	35596	1:171.5	0.0065	
第二种	驱逐:被杀	35477	1:343	0.0911	0.1423
	失败:完成	312396	1:515.5	0.0158	
第三种	驱逐:失败	325	1:2	-	-
	被杀:完成	347548	1:8.8	-	

由表1可知,四类终止状态可分为三种组合方式,首先判断三种方式整体的均衡比例,第三种方式整体并不均衡,比例为1:1069.4,而第一种和第二种方式的比例大约为1:8.77和1:8.81,所以排除第三种组合。然后根据SVDD算法,计算各个二类组合的重叠度,比较其重叠度之和,其中第二种组合方式的重叠度比第一种高了3.54%,所以排除第二种组合。因此,选择了驱逐、完成和失败、被杀的第一种二类组合方式。

3.3 欠采样参数的选择

对比实验分为两组:(1)选取欠采样过程中使用箱子个数的最优值 K ; (2)选取SP欠采样的最优比例。

(1)选取欠采样过程中箱子个数的最优值 K

SP采样不是简单地平衡正/负数据或直接分配权重,并根据硬度分布迭代地选择信息量最大的大多数数据样本。采样过程中,将数量较多类的任务分为 K 个箱子,根据硬度函数去筛选信息含量更为丰富的一个箱子。分类模型使用GBDT模型。将SP的采样比例固定在1:1,然后选取最优的 K 值,其中 K 的取值范围为[1, 15]。 K 参数的确定实验如图4所示。

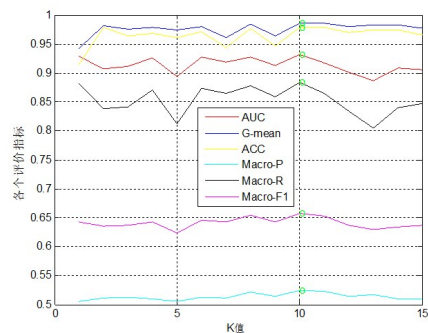


图4 不同箱子个数下各个评价指标的值

将各个评价指标画为折线图,折线图上绿色圆点为此条折线图的最高值。从图4可知,虽然随着 K 值的变化,各个评价指标的值是没有固定变化趋势,但是各个评价指标的最优值均在 $K=10$ 的时刻。其中,Macro为宏平均,是更加针对少数类预测结果正确与否的方法。综上所述,SP欠采样过程中,最终选取的 K 值为10。

(2)选取SP欠采样的最优比例

由于Google数据集中数据量大,且任务终止状态最大不均衡比例为1:3031,属于极度不均衡。如果只是按照1:1比例进行欠采样,会使得原始数据大量丢失。针对上述问题,将采样比例进行了扩展。为选取

SP的最优采样比例,本组实验从1:1递增到1:15进行测试。实验使用的多分类模型为GBDT模型,实验结果如表2所示。

欠采样的扩展比例不宜过高或过低,因为扩展比

例过高的话,数据集还会处于不均衡的状态,如果扩展比例太低,则所筛选的数据量太少。其中训练集经过SP欠采样时,扩展比例为1:1以及1:4的数据分布对比如图5所示。

表2 不同采样比例的对比实验

采样比例	Time/s	AUC	G-mean	ACC	Macro-P	Macro-R	Macro-F1
1:1	0.65	0.9039	0.9846	0.9769	0.5162	0.8225	0.6343
1:2	0.75	0.9221	0.9928	0.9893	0.6072	0.8663	0.7140
1:3	0.97	0.8741	0.9953	0.9930	0.8312	0.7648	0.7967
1:4	1.04	0.8858	0.9942	0.9912	0.9968	0.7922	0.8828
1:5	1.22	0.8981	0.9977	0.9965	0.9274	0.8045	0.8616
1:6	1.38	0.8792	0.9963	0.9945	0.8198	0.77	0.7941
1:7	1.76	0.8653	0.9972	0.9958	0.8985	0.7405	0.8119
1:8	1.78	0.8642	0.9969	0.9953	0.9982	0.7394	0.8495
1:9	1.85	0.8646	0.997	0.9955	0.9982	0.7398	0.8498
1:10	1.84	0.8796	0.9968	0.9953	0.9483	0.7704	0.8502
1:11	2.33	0.8733	0.9971	0.9957	0.8736	0.7563	0.8107
1:12	2.22	0.8655	0.9972	0.9958	0.9984	0.7407	0.8504
1:13	2.91	0.8491	0.997	0.9955	0.9982	0.7086	0.8289
1:14	3.04	0.8653	0.9971	0.9957	0.8733	0.7404	0.8014
1:15	2.87	0.8419	0.9971	0.9956	0.8891	0.6937	0.7793

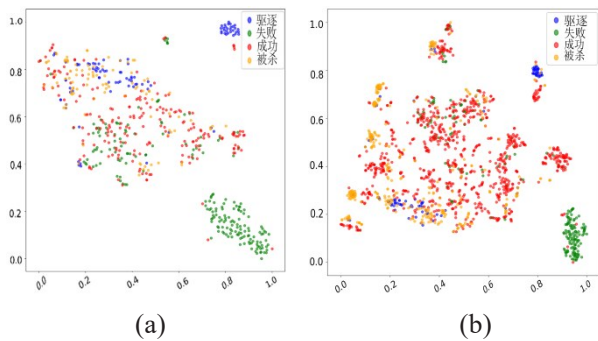


图5 SP欠采样扩展比例的数据分布对比图
(a) 1:1 (b) 1:4

从图5可以看出,采样比例为1:1时,训练集的重叠度已经明显减少,但其包含的任务数量明显较少。根据表2,选择了将1:4的扩展比例。1:1的欠采样比例扩展为1:4之后,训练集中包含的数据量明显增加,在保证筛选信息含量丰富的数据同时,也要保证原始数据不被过度删除,充分保留原有数据的底层分布特征。由表2可知,在欠采样比例为不断增加时,AUC、Macro-R、Macro-F和Macro-P的值逐渐升高后慢慢降低,G-mean、ACC和时间性能的值逐渐提升。

因此综合考虑各个指标的最优值,选取的SP欠采样比例为1:4。

3.4 多分类预测结果的分析

实验分为两组:(1)当前常用采样方法和所提方法对比,以及(2)多种分类模型和所提分类模型的对比。

采样算法:采样方法的对比采样算法有不采样、欠采样和过采样。其中5种欠采样方法分别为随机过采样(Random Over-Sampler, ROS)^[13]、自适应综合采样(Adaptive Synthesis Sampling, ADASYN)^[14],以及两个基于合成少数类的过采样技术(Synthetic Minority Over-sampling Technique, SMOTE)^[16]的改进过采样方法,即borderline-SMOTE1 (bd-SMOTE1)和borderline-SMOTE2 (bd-SMOTE2)^[17]。2种欠采样方法有Cluster Centroids (CLC)和随机下采样(Random Under Sampler, RUS)。

多分类模型:选取了两类模型,即2个单分类模型和3个集成模型进行试验。单模型有DT、SVM,集成模型有AdaBoost^[18]、Bagging^[19]、随机森林(Random Forest, RF)^[20]集成模型。其中Bagging集成方法通常考虑的是同质弱学习器,相互独立地并行学习这些弱学习器,并按照某种确定性的平均过程将它们组合起来。

(1) 采样方法的对比结果及分析

分别对数据进行不采样、欠采样、过采样和SP的1:4欠采样比例。其中各个采样方法在进行多分类时,采用分类模型为GBDT模型。

对比结果如表3所示,由于云平台数据集数据量大,过采样会导致模型的训练时间普遍较长。其中在

随机过采样之后,原本训练集的任务数量从608个变成了104362个,导致模型的训练时间显著变长。而SP的1:4欠采样训练模型所用的时间为1.04s,时间性能明显优于其他采样方法,极大地节省了云平台的调度等待时间。

表3 各种采样方法的终止状态预测结果

采样方法		Time/s	AUC	G-mean	ACC	Macro-P	Macro-R	Macro-F1
不采样	不采样	210.94	0.6955	0.9648	0.9474	0.6934	0.5171	0.5925
	ROS	501.03	0.8281	0.8875	0.8338	0.3555	0.7640	0.4852
	ADASYN	791.57	0.7812	0.8426	0.7692	0.3440	0.7060	0.4626
过采样	bd-smote1	736.86	0.7923	0.7983	0.7064	0.3140	0.7016	0.4338
	bd-smote2	764.88	0.8026	0.8177	0.7337	0.3195	0.7182	0.4423
	GAN	202.55	0.7230	0.9714	0.9573	0.7264	0.5430	0.6215
	CLC	0.48	0.6612	0.3379	0.1586	0.2917	0.5499	0.3812
欠采样	RUS	0.48	0.7932	0.7948	0.7015	0.3096	0.7263	0.4342
	SP-GBDT	1.04	0.8858	0.9942	0.9912	0.9968	0.7922	0.8828

将数据集按7:3的比例分为训练集和测试集。训练集不采样的数据分布以及训练集经过SPE的扩展比例欠采样之后,数据分布如图6所示。其中为了将高维数据可视化,将其进行了t-SNE(t-distributed stochastic neighbor embedding)降维^[21]处理。

对比图6(a)中未采样的数据分布,图6(b)中ROS的过采样方法虽然缓解了数据的不均衡程度,但数据的重叠程度依然很高,导致模型的效果并不理想。图6(c)中数据的重叠程度明显减少,但普通的欠采样方法,只是简单的将多数类样本随机剔除,被剔除的样本可能包含着一些重要信息,导致,实验效果均不理想。

由于RUS欠采样方法大量删减了信息含量较高的数据,导致欠采样后的模型进行训练时,数据缺乏,使得模型效果较差。图6(d)中可明显看出,本文方法有效降低了样本重叠度,且使得各个终止状态的样本数量保持均衡的情况下,将欠采样比例放宽到1:4,且利用硬度函数筛选信息含量丰富的任务样本,使得模型获得良好的结果。

(2)预测结果的对比与分析

在对数据集进行欠采样之后,将其输入模型中进行多分类。表4对比了各种预测模型的终止状态预测结果。其中,GBDT模型中弱学习器的最大迭代次数为20,损失函数为对数似然损失函数,决策树最大深度为默认值3。

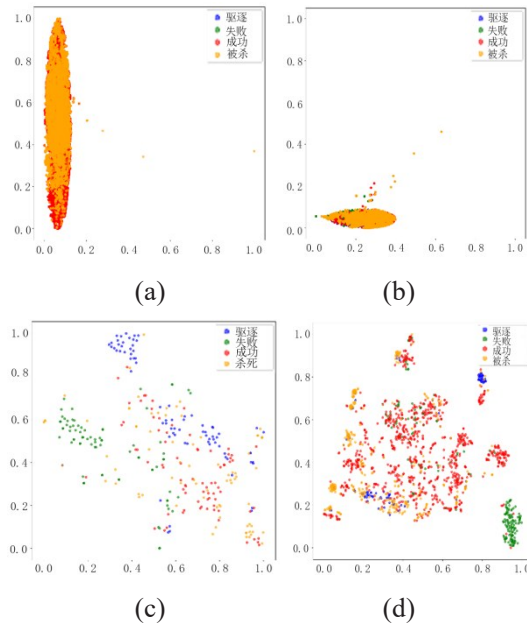


图6 经过采样方法之后的训练集数据分布 (a)未采样(b) ROS (c) RUS (d) SP-GBDT

由表4可见,在单模型中,DT模型对缺失值数据不敏感,且决策树的信息增益倾GBDT模型,其中Macro-F指标差值达到了27.68%。由于SVM模型在数据集含有较多噪声和缺失值的情况下,对缺失值非常敏感,且不能有效处理含有较多噪声的数据,因此SVM模型的各个评价指标均低于GBDT。在3种集成模型中,AdaBoost模型实现了多颗决策树的加权运算,但不能有效处理数据不均衡的分布,导致各个指标较低。由于Bagging模型

拟合程度差,使得各个指标较低。RF模型虽然能有效处理不均衡数据,但其可能具有相似的决策树,不能得到真实的多分类结果,导致其Macro-P

和Macro-F1指标较低。GBDT将gradient boosting算法和CART决策树结合在一起,其各个评价指标,在结果上基本上是较为优于各个集成模型。

表4 不同模型的终止状态预测结果比较

方法	模型	Time/s	AUC	G-mean	ACC	Macro-P	Macro-R	Macro-F1
单模型	DT	0.09	0.8654	0.9885	0.9828	0.5113	0.7438	0.6060
	SVM	7.46	0.7226	0.8978	0.8488	0.5837	0.5741	0.5789
集成模型	AdaBoost	0.67	0.7820	0.9219	0.8841	0.3660	0.5963	0.4536
	Bagging	2.70	0.8667	0.9972	0.9957	0.7004	0.7418	0.7205
	RF	0.31	0.8541	0.9949	0.9941	0.6707	0.7325	0.7002
	SP-GBDT	1.04	0.8858	0.9942	0.9912	0.9968	0.7922	0.8828

针对数据集不均衡程度和类重叠度会在一定程度上影响模型对少类预测值的问题,对少类(驱逐,失败)的预测准确度进行了进一步的分析,结果如表5所示。

表5 驱逐和失败的少类任务样本预测

	DT	SVM	AdaBoost	Bagging	RF	SP-GBDT
驱逐	0.500	0.250	0.750	0.605	0.500	0.750
失败	0.750	0.5625	0.000	0.750	0.625	0.625
均值	0.625	0.406	0.375	0.678	0.563	0.688

表5显示,在数据处于极度不均衡的状态下,SP-GBDT模型对于少数类任务样本的预测仍具有良好的准确性。

4 结束语

大型异构云计算平台上,针对任务终止状态的预测遇到数据集极度不均衡的问题,提出了一种重叠度区分的任务终止状态预测模型,且通过实验选取了最优的参数,并验证了所提方法的优越性。对于实验结果进行了深入分析。实验结果表明,SP-GBDT模型能在极大提升时间性能的基础上,获取更好的分类性能。同时,对任务的终止状态进行细粒度的多分类结果,便于云平台系统的调度决策,提升云平台的运算性能。

参考文献 (References):

- [1] Verma A, Pedrosa L, Korupolu M, et al. Large-scale cluster management at Google with Borg [C]. ACM Press the Tenth European Conference, 2015.
- [2] Jassas M, Mahmoud Q H. Failure Analysis and Characterization of Scheduling Jobs in Google Cluster Trace [C]. 2018 44th Annual Conference of the IEEE Industrial Electronics Society Washington, 2018:3102-3107.
- [3] Islam T, Manivannan D. Predicting Application Failure in Cloud: A Machine Learning Approach [C]. IEEE 1st International Conference on Cognitive Computing, 2017:24-31.
- [4] 刘春红, 韩晶晶, 商彦磊. 基于SVM分类的云集群失败作业主动预测方法 [J]. 北京邮电大学学报, 2016, 39(05): 104-109.
- [5] Padmakumari P, Umamakeswari A. Task Failure Prediction using Combine Bagging Ensemble (CBE) Classification in Cloud Workflow [J]. Wireless Personal Communications, 2019, 107(1):23-40.
- [6] Chen X, Lu C, Pattabiraman K. Failure Analysis of Jobs in Compute Clouds: A Google Cluster Case Study [C]. International Symposium on Software Reliability Engineering, 2014:167-177.
- [7] Rosa A, Chen L Y, Binder W. Failure Analysis and Prediction for Big-Data Systems [J]. IEEE Transactions on Services Computing, 2017, 10(6):984-998.
- [8] Chen X, Lu C, Pattabiraman K. Failure Prediction of Jobs in Compute Clouds: A Google Cluster Case Study [C]. IEEE International Symposium on Software Reliability Engineering Workshops, 2014:341-346.
- [9] Liu C H, Dai L P, Lai Yi, et al. Failure prediction of tasks in the cloud at an earlier stage: a solution based on domain information mining [J]. Computing, 2020: 2001-2023.
- [10] Zhang Q, Zhan M F, Boutaba R, et al. Dynamic Heterogeneity-Aware Resource Provisioning in the Cloud [J]. IEEE Transactions on Cloud Computing, 2014, 2(1):14-28.
- [11] Sanchez-Hernandez C, Boyd D S, Foody G M. One-Class Classification for Mapping a Specific Land-Cover Class: SVDD Classification of Fenland [J]. IEEE Transactions on Geoscience & Remote Sensing, 2007, 45(4):1061-1073.
- [12] Liu Z, Cao W, Gao Z, et al. Self-paced Ensemble for Highly Imbalanced Massive Data Classification [C]. International

- al Conference on Data Engineering (ICDE), 2020.
- [13] Moreo A, Esuli A, Sebastiani F. Distributional Random Oversampling for Imbalanced Text Classification [C]. the 39th International ACM SIGIR conference, 2016:805-808.
- [14] He H, Yang B, Garcia E A, et al. ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning [C]. IEEE International Joint Conference on Neural Networks, 2008:1322-1328.
- [15] Wang K F, Gou C, Duan Y J, et al. Generative Adversarial Networks: The State of the Art and Beyond[J]. Zidonghua Xuebao/acta Automatica Sinica, 2017, 43(3):321-332.
- [16] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: Synthetic Minority Over-sampling Technique [J]. Journal of Artificial Intelligence Research, 2002, 16(1):321-357.
- [17] Han H, Wang W Y, Mao B H. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning [J]. Advances Intelligent Computing, 2005:878-887.
- [18] Yang S, Chen L F, Yan T, et al. An ensemble classification algorithm for convolutional neural network based on Ada-Boost [C]. IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), 2017:401-406.
- [19] Razzaghi T, Xanthopoulos P, Şeref O. Constraint relaxation, cost-sensitive learning and bagging for imbalanced classification problems with outliers [J]. Optimization Letters, 2015, 11(5):915-928.
- [20] Shen J, Wan J, Lim S J, et al. Random-forest-based failure prediction for hard disk drives [J]. International Journal of Distributed Sensor Networks, 2018, 14(11).
- [21] Arora S, Hu W, Kothari P K. An Analysis of the t-SNE Algorithm for Data Visualization [J]. Proceedings of Machine Learning Research, 2018, 75:1-8.

编辑:龙学锋,李树锋

(上接第43页)

- [27] House F. Freedom on the Net 2017: Manipulating Social Media to Undermine Democracy [R/OL]. Freedom House, <https://freedomhouse.org/report/freedom-net/freedom-net2017>, 2017—11.
- [28] Howard P, & Kollanyi B. Bots. #StrongerIn, and #Brexit: Computational propaganda during the UK-EU Referendum [R]. COMPROP Research Note, 2016.
- [29] Irani D, Webb S, & Pu C. Study of static classification of social spam profiles in myspace [A]. In Fourth International AAAI Conference on Weblogs and Social Media, 2010.
- [30] Jr S B, Campos G F C, Tavares G M, Igawa R A, Jr M L P, & Guido R C. Detection of human, legitimate bot, and malicious bot in online social networks based on wavelets [J]. ACM Transactions on Multimedia Computing Communications and Applications, 2018, 14(1): 1-17.
- [31] Kevin S. Xu, David Reitter, Dongwon Lee, Nathaniel Osgood. Social, Cultural, and Behavioral Modeling. Washington, DC, USA: Springer, 2016.
- [32] Khaldarova I, Pantti M. Fake news: The Narrative Battle Over the Ukrainian Conflict [J]. Journalism Practice, 2016, 10(7): 891-901. (<https://www.oii.ox.ac.uk/research/projects/computational-propaganda/>, 2016—01—01)
- [33] Lasswell H D. Propaganda technique in the world war. Ravenio Books, 1927.
- [34] Lasswell H D. Propaganda In ERA [A]. Seligman & A Johnson (Eds). Encyclopedia of the social sciences [M]. New York Macmillan, 1937, 12.521-528.
- [35] Lau R Y, Liao S Y, Kwok R C W, Xu K, Xia Y, & Li Y. Text mining and probabilistic language modeling for online review spam detection [J]. ACM Transactions on Management Information Systems (TMIS), 2012, 2(4): 1-30.
- [36] Loyola-González O, Monroy R, Rodríguez J, López-Cuevas A, & Mata-Sánchez J I. Contrast pattern-based classification for bot detection on Twitter. IEEE Access [J], 2019, 7: 45800-45817.
- [37] Michael K. Bots without borders: How anonymous accounts hijack political debate [EB/OL]. <http://theconversation.com/bots-without-borders-how-anonymous-accounts-hijack-political-debate-70347>, accessed 18 July 2019.
- [38] Northcutt C G, Jiang L, Chuang I L. Confident Learning: Estimating Uncertainty in Dataset Labels [J]. 2019.
- [39] Oliveira et al. The Influence of Retweeting Robots During Brazilian Protests [A]. 49th Hawaii International Conference on System Sciences, 2016.
- [40] Subrahmanian V S, Azaria A, Durst S, Kagan V, Galstyan A, Lerman K, ... & Menczer F. The DARPA Twitter bot challenge [J]. Computer, 2016, 49(6), 38-46.
- [41] Thomas Renault. Market manipulation and suspicious stock recommendations on social media [EB/OL]. <http://www.thomas-renault.com/wp/market-manipulation-suspicious.pdf>
- [42] Yang K C, Varol O, Davis C A, Ferrara E, Flammini A, & Menczer F. Arming the public with artificial intelligence to counter social bots [J]. Human Behavior and Emerging Technologies, 2019, 1(1), 48-61.

编辑:龙学锋,李树锋