软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2015,26(11):2795−2810 [doi: 10.13328/j.cnki.jos.004897]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一种大数据环境中分布式辅助关联分类算法*

张明卫[1], 朱志良[1], 刘 莹[1], 张 斌[2]

[1](东北大学 软件学院,辽宁 沈阳 110004)

[2](东北大学 信息科学与工程学院,辽宁 沈阳 110004)

通讯作者: 张明卫, E-mail: zhangmw@swc.neu.edu.cn

摘 要: 在很多现实的分类应用中,新数据的类标需要由领域专家最终确定,而分类器的分类结果仅起辅助作用.另外,随着大数据所隐含价值越发被人们重视,分类器的训练会从面向单一数据集逐渐过渡到面向分布式空间数据集,大数据环境下辅助分类也将成为未来分类应用的重要分支.然而,现有的分类研究缺乏对此类应用的关注.大数据环境中的辅助分类面临以下 3 个问题:1) 训练集是分布式大数据集;2) 在空间上,训练集所包含的各局部数据源的类别分布不尽相同;3) 在时间上,训练集是动态变化的,会发生类别迁移现象.在考虑以上问题的基础上,提出一种大数据环境中分布式辅助关联分类方法.该方法首先给出一种大数据环境中分布式关联分类器构建算法,在该算法中,通过横向加权考虑分类数据集在空间上的类别分布差异,并给出"前件空间支持度-相关系数"的度量框架,改进关联分类算法面对不平衡数据的性能缺陷;然后,给出一种基于适应因子的辅助关联分类器动态调整方法,能够在分类器应用过程中充分利用领域专家实时反馈的结果对分类器进行动态调整,以提升其面向动态数据集的分类性能,减缓分类器的退化和重新训练的频率.实验结果表明,该方法能够面向分布式数据集较快地训练出有较高分类准确率的关联分类器,并在数据集不断扩充变化时提升分类性能,是一种有效的大数据环境中辅助分类应用方法.

关键词: 大数据;分布式;辅助分类;关联分类;动态分类器

中图法分类号: TP311

中文引用格式: 张明卫,朱志良,刘莹,张斌.一种大数据环境中分布式辅助关联分类算法.软件学报,2015,26(11):2795−2810. http://www.jos.org.cn/1000-9825/4897.htm

英文引用格式: Zhang MW, Zhu ZL, Liu Y, Zhang B. Distributed assistant associative classification algorithm in big data environment. Ruan Jian Xue Bao/Journal of Software, 2015,26(11):2795−2810 (in Chinese). http://www.jos.org.cn/1000-9825/4897.htm

Distributed Assistant Associative Classification Algorithm in Big Data Environment

ZHANG Ming-Wei[1], ZHU Zhi-Liang[1], Liu Ying[1], ZHANG Bin[2]

[1](Software College, Northeastern University, Shenyang 110004, China)

[2](College of Information Science and Engineering, Northeastern University, Shenyang 110004, China)

Abstract: For many practical classification applications, the class label of new data needs to be confirmed eventually by domain expert, and the result of classifier only plays an assistant role. In addition, with the implicit values of big data calling more people's attention, classifier training is going through a transition from single dataset to distributed space dataset, and assistant classification in big data environment will also become an important branch of future classification applications. However, existing classification research lacks attention to this kind of application. Assistant classification in big data environment faces with the following three problems: 1) the training set is distributed big dataset, 2) in space, the class distributions of local datasets contained in the training set are commonly different, and 3) in time, the training set is dynamic and its class distribution may change. To address the above problems, this paper

* 基金项目: 国家自然科学基金(61100027, 61374178, 61202085, 61572117, 61572116); 中央高校基本科研业务费专项资金(N130417003); 高等学校博士学科点专项科研基金(20120042120010)

收稿时间: 2015-05-27; 修改时间: 2015-07-14, 2015-08-11; 定稿时间: 2015-08-26

proposes a distributed assistant associative classification approach in big data environment. Firstly, a distributed associative classifier constructing algorithm in big data environment is constructed. With the new algorithm, the class distribution difference in space of the classification dataset is considered by horizontal weighting, and the performance deficiency of associative classification algorithms to imbalanced class distribution datasets is improved by giving a measure framework of "antecedent space support-correlation coefficient". Next, an adaptive factor based dynamic adjustment method for assistant associative classifier is proposed. This method can make full use of domain experts' real-time feedback to adjust classifier dynamically in the applying process of the used classifier, to improve its performance facing dynamic datasets, and to slow down its retraining frequency. Experimental results demonstrate that the presented approach can relative quickly train associative classifiers with higher classification accuracy for distributed datasets, and can improve their performance when datasets are continually expanding and changing. Thus it's an effective approach for assistant classification applications in big data environment.

Key words: big data; distribution; assistant classification; association classification; dynamic classifier

作为数据挖掘的主要任务之一,分类已得到了广泛的研究与应用[1].然而在很多现实分类应用中,挖掘出的分类器仅对新采集的未知类标的数据进行辅助类别判定,而该实例的类标最终由领域专家参考自动分类结果来确定.称此类应用为辅助分类,其在医疗诊断、欺骗侦测等许多领域使用较为广泛.另外,随着信息技术的快速发展,数据呈现出爆炸式的增长趋势.传统的基于单机的数据仓库已不能满足海量数据的存储要求,而具有良好扩展性的分布式数据仓库逐渐成为各行业数据存储的新选择[2].相应的分类挖掘也会从面向单一数据集过渡到面向分布式空间数据集.因而,大数据环境下的辅助分类将成为一个有价值的分类应用分支,但现有的分类研究缺乏对它的关注.

在大数据环境下的辅助分类应用中,数据集是分布式存放、实时采集、不断扩充的.以中医领域对小儿肺炎疾病的诊治为例,为了提升诊治水平,可以促进全国病例数据的分布式共享与应用[3].基于此训练出的分类器虽然能够判别新采集病例所属的证候,比如说是痰热闭肺、风热闭肺、气阴两虚或者是其他证候等,以判别该病例的发病主因,并对其进行"去跟"治疗.但为了确保疾病诊治效果,新病例的证候还是应该由中医专家最终确定.在此类错分代价比较敏感的应用中,并不是说挖掘出的分类器不重要,相反,一个良好的分类器应该在绝大多数情况下和领域专家的意见相同,当出现少数和专家意见不同的数据时,可以促使专家重视以提升分类应用效果.

如何针对大数据背景给出一种高效的辅助分类器构建算法,面临着以下 3 个方面的问题:首先,分类器的构建要适应大数据的要求,能够在分布式空间数据集上有效地训练出分类器;其次,在训练分类器时,要充分考虑分类数据集在空间上的类别分布差异,以中医小儿肺炎证候分类为例,在黑龙江、辽宁、山东、四川、上海和广东等各点上所采集病例的类别分布不尽相同,因而不易构建统一的全局分类器,而应该针对各点,侧重考虑本地数据集,以其他点上的数据集作为有机补充,构建适应于各点类别分布特征的分类器;此外,在训练分类器时还要考虑分类数据集随时间所发生的类别迁移现象[4],比如中医小儿肺炎的证候类别分布会随季节的更替而发生变化.这就使得随着新分类数据的不断加入,由原先训练样本集构造出的分类器在分类准确度上会逐渐下降,从而需要不断重复地更新样本集并训练出新的分类器,以提高分类准确度.但据我们的最新查询结果,当前,分类方法构造出来的均是静态分类器.即,在下次重新训练之前,分类器在应用过程中保持不变.这使得分类器不能在分类过程中充分利用领域专家给出的实时反馈结果,在重新训练前不能自动调整以适应新的数据,从而在一定程度上影响了分类器面向动态数据集的分类性能.

在综合考虑以上 3 个问题的基础上,本文提出一种大数据环境中分布式辅助关联分类算法(distributed assistant associative classification algorithm in big data environment,简称 AAC).关联分类是近年来发展起来的一种基于关联规则[5]的分类方法,其基本思想是:通过频繁模式的搜索产生分类关联规则,然后对规则进行剪枝并形成分类器,经典算法包括 CBA[6],CMAR[7],CPAR[8],CAEP[9]和 HARMONY[10]等.这类方法的出现虽然比决策树方法[11]、贝叶斯分类方法[12]、神经网络方法[13]、*K*-近邻方法[14]等相对较晚,但由于充分利用了挖掘出的数据属性和类标签之间的强关联知识,其往往可以达到比其他类型分类器更高的分类准确度,也因此成为分类研究与应用领域的一个重要和有价值的分支.本文首先针对大数据背景给出一种分布式关联分类器构建算法,其中,

通过横向加权和定义“空间支持度”和“空间置信度”的概念来考虑分类数据集在空间上的类别分布差异;通过给出“强前件相关规则”的定义,以“前件空间支持度-相关系数”代替“支持度-置信度”的度量框架,以此弥补关联分类器面对不平衡数据的性能缺陷.接下来,本文给出一种基于适应因子的辅助关联分类器动态调整方法,能够在分类过程中充分利用领域专家的反馈结果,并动态调整分类器自身,以提升其面向动态数据集的分类性能,减缓分类器的退化和重新训练的频率.实验结果表明:本文方法能够面向分布式数据集较快地训练出有较高分类准确率的关联分类器,并在数据集不断扩充变化时提高分类性能,是一种有效的大数据环境中分布式辅助分类应用方法.

本文主要有以下贡献点:首先,本文分析了大数据环境中辅助分类问题,考虑了大数据环境下分类数据集在空间和时间上的类别分布特征变化以及分类过程中领域专家对分类结果的实时反馈,提出了一套基于关联分类的大数据环境下辅助分类应用的解决方案,可为以后此类问题的研究提供原型参考;其次,本文提出了一种大数据环境中分布式关联分类器构造算法,能够在着重考虑本地数据集的基础上,为各点训练出有较高分类准确率的分类器;最后,本文提出了一种辅助分类器的动态调整算法,能够在分类过程中根据专家的实时反馈结果动态调整分类器自身,提高其面向新数据的适应能力.

本文第 1 节将描述本文大数据环境中分布式辅助关联分类模型.第 2 节描述大数据环境中分布式关联分类器的构建算法.第 3 节将给出辅助关联分类器的动态调整方法.第 4 节针对文中提出的大数据环境中辅助关联分类方法进行实验分析.第 5 节进行总结,并展望今后的工作.

1 大数据环境中分布式辅助关联分类模型

随着大数据应用的不断延伸,分类会逐渐由面向单一数据集过渡到面向分布式空间数据集.另外,很多分类应用往往以分类器为辅、领域专家为主交互完成.在大数据环境下,辅助分类应用中存在着大数据的分布性、数据集在空间上类别分布的差异性和数据集在时间上类别分布的迁移性这三大特征.现有的分类研究还未充分重视该类应用及应用中所面临的问题,为此,本文提出了一种大数据环境中分布式辅助关联分类方法,其模型如图 1 所示,分为分类器挖掘和分类器应用两个阶段,主要包含大数据环境中分布式关联分类器的构建、应用过程中辅助关联分类器的动态调整以及分类器重训练的触发这 3 部分主要内容.

- 大数据环境中分布式关联分类器的构建

在大数据环境中,用于分类的数据仓库不再是集中、单一的,而是由分布在不同空间中的各局部数据源组成.然而,每个局部数据源都有着自身的类别分布特征,其他局部数据源与其在类别分布特征上的相似程度也不尽相同.因而在大数据环境中进行分类时,需要针对各局部数据源训练出不同的分类器.选定某一局部数据源为本地数据源,在为其训练分类器时要重点考虑它本身的类别分布特征,同时,其他局部数据源又是其有机补充,从而提升训练出的分类器在本地数据源上的分类性能.为此,本文提出一种大数据环境中分布式关联分类器构建算法.该算法首先通过横向加权,在分布式空间数据集上构建一棵重点考虑本地数据源的全局加强 FP-tree,在此基础上,考虑关联分类面向类别分布不平衡数据集的性能缺陷,以“前件空间支持度-相关系数”为度量框架,生成一棵分类规则树 CR-tree,即为本地数据源在整个空间数据集上训练出的最终分类器.可以调用该算法为各局部数据源生成各自的分类器;或者当分类器在应用过程中性能下降需要重新训练时,调用该算法在动态数据集上训练出新的分类器.这部分内容将在第 3 节详细加以介绍.

- 应用过程中辅助关联分类器的动态调整

在一些分类的实际应用中,分类数据集往往是实时采集、不断扩充的.良好的分类器应该能够较快地适应新的分类数据并做出较为准确的判别,以辅助专家对新数据进行类别判定.现有的分类方法缺乏对辅助分类应用中专家实时反馈结果的考虑,使训练出来的往往是面向动态数据集退化较快的静态分类器.为此,本文给出一种辅助关联分类器动态调整算法,能够基于挖掘出的 CR-tree 预测新的待分类数据的类标;同时,能够根据领域专家反馈的实时分类结果,对分类器进行动态自适应调整,以使分类器在分类过程中加快适应新的分类数据,提高分类准确率.这部分内容将在第 4 节中详细加以描述.

- 基于分类结果统计的分类器重训练的触发

随着新的分类数据的不断加入以及可能带来的类别迁移等现象,即使在分类过程中,动态调整的分类器也会出现性能下降而需要进行重新训练.本文进行分类器重训练的触发条件是:$(n>N)\wedge(n_1/n<Acc)$,其中,n 为新分类数据条数,N 为重训练数据个数阈值,n_1 是 n 条新数据中分类器正确分类的条数,Acc 是重训练准确率阈值.即,当分类器已应用的实例数大于 N 且准确率低于阈值 Acc 时,将触发本文辅助关联分类器的挖掘算法,以重新训练分类器.

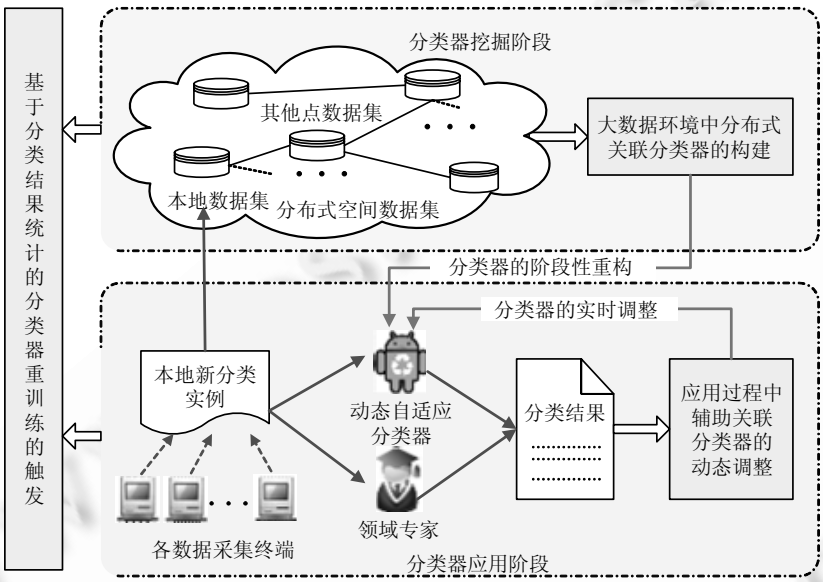


Fig.1 Distributed assistant associative classification model in big data environment

图 1 大数据环境中分布式辅助关联分类模型

2 大数据环境中分布式关联分类器的构建

要构建大数据环境下的分布式关联分类器,其主要工作是在分布式训练集上挖掘产生用于分类的类关联规则集.目前,分布式关联规则挖掘已取得较多成果,如基于 Apriori 思想的 CD[15],FMD[16],DDDM[17]等算法和基于 FP-growth[18]思想的 MLFPT[19],FDMA[20],FMAGF[21],DMARF[22]等算法.两种思路相比,前者存在候选项集多、通信量大、同步次数多和扫描数据库次数多等问题.

与普通分布式环境相比,在大数据分类应用中,数据集往往分布在较不稳定的广域网环境中,因此,各局部数据源间的通信次数在很大程度上决定了算法执行的稳定性和性能.为了能够在大数据环境中训练出高效的分类器,本文提出了一种大数据环境中分布式关联分类器构建方法.与上述方法对比,各局部数据源间的总体通信次数最少.下面将对该方法进行详述.

2.1 相关概念

设 D 是一个空间数据集(对应于大数据环境中用于分类的数据仓库),由 n 个分布在不同站点的点数据集(对应于各局部数据源)$\{D_0,D_1,\ldots,D_{n-1}\}$ 组成.不失一般性,以 D_0 作为本地数据集.本文基于横向加权的方法来考虑不同点数据集的类别分布特征.设 $\{w_0,w_1,\ldots,w_{n-1}\}$ 分别是各点数据集对应的分类重要度权值.当针对 D_0 训练分类器时,如果 D_i 与 D_j 相比,其类别分布特征与 D_0 更相似,则应有 $w_i>w_j$.各点数据集的分类重要度权值一般由领域专家根据经验给出.

设 D 中的每个元组由 k 个特征属性 $\{A_1,A_2,\ldots,A_k\}$ 和一个类标号属性 C 描述.$|D|$ 为空间数据集 D 中元组的个数.项 i 是一个属性值对,即某个属性 A_i 及其在该属性上的取值 a_i 的组合,记作 $\langle A_i,a_i\rangle$.项集 $I=\{i_1,i_2,\ldots,i_j\}$ 是一个由

项组成的集合.

定义 1(空间支持度(space support,简称 ss)). 设项集 I 在空间数据集 D 包含的各点数据集$\{D_0,D_1,\ldots,D_{n-1}\}$上分别出现的频数为 $f_0,f_1,\ldots,f_{n-1}$,则项集 I 在空间数据集 D 上的空间支持度为

$$ss(I)=\sum_{i=0}^{n-1}w_i f_i \Big/ \sum_{i=0}^{n-1}w_i\,|D_i| \tag{1}$$

由以上定义可知,空间支持度 $ss\in[0,1]$,它描述了项集 I 在给定空间数据集 D 中出现的频繁程度.如果项集 I 的空间支持度 ss 大于最小支持度阈值(min_ss),则称 I 为频繁项集.

关联分类方法利用关联规则挖掘算法训练出用于分类的特殊关联规则,此类规则称为类关联规则.其前件为普通的项集,后件为一个类标号,形如 $I\Rightarrow c_i$.如果元组 d_i 包含了项集 I,则称为元组 d_i 匹配规则 $I\Rightarrow c_i$.两个规则 r_1:$I\Rightarrow c_i$ 和 r_2:$I'\Rightarrow c_i'$,如果 $I\subset I'$,则称 r_1 是 r_2 的泛化规则.

定义 2(空间置信度(space confidence,简称 sc)). 类关联规则 $I\Rightarrow c_i$ 在空间数据集 D 中的置信度定义为

$$sc(I\Rightarrow c_i)=ss(I\cup c_i)/ss(I) \tag{2}$$

由定义 2 可知,空间置信度 $sc\in[0,1]$.它描述了在空间数据集 D 中包含项集 I 的元组里同时包含 c_i 的比率,确定了规则 $I\Rightarrow c_i$ 的可信程度.同时满足最小空间支持度阈值(min_ss)和最小空间置信度阈值(min_sc)的类关联规则称为强类关联规则.关联分类算法首先需要在大型空间数据集中发现强类关联规则.

2.2 面向空间数据集的加强FP-tree的构建

关联分类器构建的主要工作是产生用于分类的类关联规则集.为了使规则挖掘具有较高的缩放性和效率,能够适应大数据的要求,类似于集中式关联分类挖掘算法 CMAR,将关联分类器的构建分为两步:第 1 步构造一个高度压缩的数据结构——全局加强 FP-tree,以代表原分布式空间数据集,并在其中保存满足每个频繁项集的元组的类分布;第 2 步在加强 FP-tree 上挖掘产生并筛选类关联规则集,构造 CR-tree 以用于分类.

接下来,将就第 1 步结合下面例子描述面向空间数据集的分布式加强 FP-tree 的构建方法.

例 1:给定一空间数据集 D,由 3 个分布式点数据集$\{D_0,D_1,D_2\}$组成,见表 1~表 3.

Table 1 Distributed dot dataset D_0

表 1 分布式点数据集 D_0

Row-id	A	B	C	D	Class label
1	a_1	b_1	c_1	d_1	X
2	a_1	b_2	c_2	d_2	X
3	a_2	b_2	c_1	d_3	Y
4	a_2	b_2	c_3	d_4	Y
5	a_1	b_3	c_1	d_4	Z

Table 2 Distributed dot dataset D_1

表 2 分布式点数据集 D_1

Row-id	A	B	C	D	Class label
1	a_1	b_2	c_3	d_4	Z
2	a_1	b_1	c_2	d_4	X
3	a_3	b_2	c_1	d_3	Y
4	a_1	b_3	c_1	d_2	X

Table 3 Distributed dot dataset D_2

表 3 分布式点数据集 D_2

Row-id	A	B	C	D	Class label
1	a_1	b_1	c_1	d_1	X
2	a_1	b_2	c_2	d_4	Z
3	a_3	b_2	c_3	d_4	Y

选 D_0 为本地数据集,对应的各点数据集分类重要度权值分别为$\{w_0=1,w_1=0.5,w_2=0.7\}$,给定空间支持度阈值 min_$ss$=0.3,则为 D_0 构建加强 FP-tree 的过程如下:

1) 扫描空间数据集 D 一次.统计各点数据集 D_i 中各项出现的频率,并计算它们在 D 中的空间支持度,对各频繁项按空间支持度降序排序,结果为频繁项表 L.以项$\langle A,a_1\rangle$为例,在点数据集 D_0~D_2 出现的次数分别为 3,3,2.因此,该项在空间数据集 D 中的空间支持度为(3×1+3×0.5+2×0.7)/(5×1+4×0.5+3×0.7)=0.65.例 1 中,各项的空间支持度计算见表 4.基于设定的空间支持度阈值 0.3,得频繁项表:

$$L=\{a_1,b_2,c_1,d_4\}.$$

2) 对 D 中的每个点数据集 D_i,创建所对应的局部加强 FP-tree FPT_i,执行:a) 创建 FPT_i 的根节点,以“root”标记;b) 扫描 D_i 一次,将 D_i 中的每个元组 *tuple* 如下处理:选择 *tuple* 中的频繁项,并按 L 中的次序排序;将排序后的 *tuple* 中的频繁项表插入到树 FPT_i 中,并标记类标号.

3) 将各局部加强 FP-tree FPT_i 传输到本地机 D_0 上并进行合并,最终生成的树 *FPT* 就是为本地数据集 D_0 所构建的加强 FP-tree.

Table 4 Frequency and space support of all items

表 4 各项的频率及空间支持度

Item	a_1	a_2	a_3	b_1	b_2	b_3	c_1	c_2	c_3	d_1	d_2	d_3	d_4
D_0	3	2	0	1	3	1	3	1	1	1	1	1	2
D_1	3	0	1	1	2	1	2	1	1	0	1	1	2
D_2	2	0	1	1	2	0	1	1	1	1	0	0	2
Space support	**0.65**	0.22	0.13	0.24	**0.59**	0.16	**0.52**	0.24	0.24	0.19	0.16	0.16	**0.48**

按照以上处理过程,得到例 1 中 D_0 所对应的加强 FP-tree,如图 2 所示.

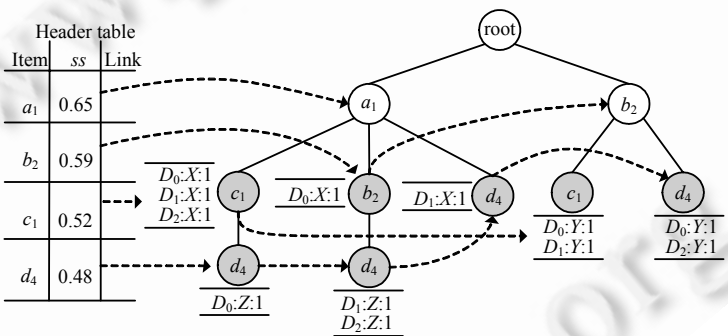


Fig.2 Corresponding strong FP-tree of D_0

图 2 D_0 所对应的加强 FP-tree

下面就面向空间数据集的分布式加强 FP-tree 的构建算法 FPT-Constructing 描述如下:

算法 1. FPT-Constructing.

输入:空间数据集 $D=\{D_0,D_1,\ldots,D_{n-1}\}$;

各点数集分类重要度权值$\{w_0,w_1,\ldots,w_{n-1}\}$;

最小空间支持度阈值 min_*ss*.

输出:本地数据集 D_0 所对应的加强 FP-tree *FPT*.

```
L=Frequent(D);                          //查找 D 中所有频繁 1 项集并排序
Broadcast(L);                           //将 L 广播到各点数据集 Di 上
FOR (Di∈D){
  FPTi=Initiate();                      //创建初始的局部加强 FP-tree FPTi
  FOR (tj∈Di) {                         //对于 Di 中的任一元组 tj
    P=Frequent(tj,L);                   //生成 tj 中包含的按 L 排序的频繁项表 P
    Insert(P,FPTi);                     //递归地将 P 插入 FPTi 中
```

```
    }
    Transfer(FPT_i,D_0);                    //将 FPT_i 传输到本地机 D_0 上
  }
  FPT=Initiate ();                          //创建初始全局加强 FP-tree FPT
  FOR (∀FPT_i)
    Merge(FPT_i,FPT);                       //将各 FPT_i 合并到全局加强 FP-tree FPT 中
  RETURN FPT;
  Procedure. Insert(P,Tree)
  p=First(P);                               //p 是 P 中第 1 个元素
  P′=P–p;                                   //P′是剩余元素表
  IF (p==null) return;
  IF (N=OneOfChind(p,Tree)) {               //如果 p 与 Tree 的某个子女 N 是同一个频繁项
    IF (P′==null) AddClassLable(N);         //如果 p 是最后一个频繁项,则为 p 所在节点 N 添加类标号信息
    Insert(P′,N);                           //将剩余频繁项表 P′插入树 N 中
  }
  ELSE {                                    //如果 Tree 不包含与 p 相同的子女,即,N 为 NULL
    N=CreateNode(p);                        //为 p 创建新结点 N
    IF (P′==null) AddClassLable(N);
    LinkToTree (N,Tree);                    //将 N 链接到它的父节点 Tree
    LinkToList(N);                          //将 N 通过节点链结构链接到具有相同项的节点上
    Insert(P′,N);
  }
```

以上描述了为单一本地数据集在整个空间数据中构建分布式加强 FP-tree 的算法 FPT-Constructing.可以通过简单的改进,为所有的点数据集构建自己的分布式加强 FP-tree,这里不再详述.设空间数据集 D 中包含 n 个点数据集,要构建全局加强 FP-tree,仅需在网络中传输 $4n$ 次,分别用于广播算法开始命令、传输局部频繁 1 项集、广播全局频繁 1 项集和传输各局部加强 FP-tree.设 D 中元组个数为 k,加强 FP-tree 树的高度为 h,则算法的时间复杂度为 $O(kh+n)$.因为构建加强 FP-tree 仅需扫描两遍空间数据集 D,因此,算法 FPT-Constructing 有着较高的运行效率.

2.3 分布式关联分类器的构建

在为点数据集 D_0 构建完分布式加强 FP-tree FPT 后,即可在 FPT 上挖掘产生类关联规则,并对规则进行剪枝,以生成分类规则树 CR-tree,用于最终分类.相对于第 1 步分布式加强 FP-tree 的构建,第 2 步分类规则树 CR-tree 的生成较为简单.

基于第 1 步得到的频繁项表 L 和分布式加强 FP-tree FPT,采用 FP-growth 算法的变形来发现满足最小空间支持度和最小空间置信度阈值的类关联规则集,即,以 L 自底向上的次序在 FPT 中产生互不重叠的类关联规则.以例 1 为例,将依次挖掘得到:1) 包含 d_4;2) 包含 c_1 但不包含 d_4;3) 包含 b_2 但不包含 c_1 和 d_4;4) 仅包含 a_1 的类关联规则集.

然而,分类应用中数据集的类别分布往往是不平衡的,即,存在某些类别的实例数特别少或某些类别的实例数特别多的情况,这在一定程度上降低了关联分类方法的分类准确率.主要有两个原因:

- 首先,因为不平衡数据集中少数类的支持度本身很低,如果将支持度阈值设置较高,则很难挖掘到支持少数类别的关联规则;而如果为得到少数类别的规则将支持度阈值设置较低,则会产生大量的其中包含许多冗余和噪声的多数类的规则.特别是本文面向分布式大数据的关联分类方法,如果将支持度阈值设置较低,则势必会产生较大的需要在各点上进行传输的加强 FP-tree,从而会在一定程度上降低本

文分类器训练的性能.

- 其次,不平衡分类数据集中可能存在大类别数据,即,存在大类别 c_i,其空间支持度为 $ss(c_i)$,本身就很大,如 0.9,如果当前挖掘出一条类关联规则 $I\Rightarrow c_i$,那么,虽然其具有较高的空间置信度 0.89,但此时前件 I 的出现并没有推进类别 c_i 出现的概率,甚至出现了一定的抑制作用.显然,该规则是无用的,甚至是具有误导性的.因此在关联分类方法中,基于支持度-置信度框架对不平衡数据集进行分类是具有局限性的.

数据采样、代价敏感学习、boosting 技术、核方法、主动学习以及单类别学习等方法是处理不平衡数据的常见策略[23−25],但这些方法在分类器训练过程中往往需要针对不平衡数据进行额外的处理工作,本文则直接通过构建合适的关联分类模型来适应不平衡数据集的分类需求.为了提升本文关联分类算法的训练性能,同时提高分类准确率,给出了强前件类相关规则的概念.

定义 3(前件空间支持度(antecedent space support,简称 ass)). 类关联规则 $I\Rightarrow c_i$ 在空间数据集 D 中的前件空间支持度定义为:$ass(I\Rightarrow c_i)=ss(I)$,即规则 $I\Rightarrow c_i$ 的前件 I 在空间数据集 D 中的空间支持度.

定义 4(相关系数(correlation coefficient,简称ρ)). 给定空间数据集 D 上的一个类关联规则 $I\Rightarrow c_i$,定义其前件 I 和后件 c_i 间的相关系数如下:

$$\rho(I\Rightarrow c_i)=\frac{ss(I\cup c_i)-ss(I)ss(c_i)}{\sqrt{ss(I)(1-ss(I))ss(c_i)(1-ss(c_i))}} \tag{3}$$

规则 $I\Rightarrow c_i$ 的相关系数 $\rho(I\Rightarrow c_i)$ 取值范围代表的含义分别是:1) 当 $\rho(I\Rightarrow c_i)\in(0,1)$ 时,项集 I 的出现将对类别 c_i 的出现起推进作用,称 $I\Rightarrow c_i$ 为正相关规则;2) 当 $\rho(I\Rightarrow c_i)=0$ 时,项集 I 的出现与类别 c_i 的出现之间没有任何联系,称其为独立规则;3) 当 $\rho(I\Rightarrow c_i)\in[-1,0)$ 时,项集 I 的出现将对类别 c_i 的出现起抑制作用,称其为负相关规则.显然,规则前件 I 对后件 c_i 的出现所起的推进或抑制作用越明显,即,其相关系数的绝对值越接近于 1,其对分类的重要度就越大.

定义 5(强前件类相关规则(strong antecedent class correlation rule,简称 saccr)). 在空间数据集 D 中,给定最小空间支持度阈值 min_ss 和最小相关系数阈值 min_ρ,如果 $ass(I\Rightarrow c_i)\geqslant$min_$ss$ 且 $|\rho(I\Rightarrow c_i)|\geqslant$min_$\rho$,则称规则 $I\Rightarrow c_i$ 为空间数据集 D 中的强前件类相关规则.

与以往的关联分类算法不同,本文在加强 FP-tree FPT 上发现强前件类相关规则用于分类.由于在规则发现时考虑的是满足最小空间支持度阈值的前件与各类别间的相关系数,所以可使不平衡数据集中各类别发现规则的机会相对均等,同时,分类时又能充分利用规则前件对后件的推进和抑制两方面的信息.下面结合例 1 说明分类规则树 CR-tree 的生成过程.给定例 1 中的最小空间支持度阈值 min_ss=0.3 和最小相关系数阈值 min_ρ=0.3.

首先,在图 2 所示的加强 FP-tree FPT 中挖掘包含 d_4 的类相关规则集.构造 d_4 的条件模式基(一个子数据库,由 FPT 中与 d_4 一起出现的前缀路径集组成),如图 3 所示.以项 b_2 为例,给定 d_4 的条件模式基(与 d_4 一同出现),其在 D 中的空间支持度为 $(w_0\times1+w_1\times1+w_2\times2)/(w_0\times|D_0|+w_1\times|D_1|+w_2\times|D_2|)=0.32$.在条件模式基中删掉不满足空间支持度阈值的项 c_1,得 d_4 的条件加强 FP-tree 如图 4 所示.由此可产生频繁模式:$(a_1,d_4)|ss=0.30$ 和 $(b_2,d_4)|ss=0.32$.而 (a_1,b_2,d_4) 的空间支持度为 $(w_1\times1+w_2\times2)/(w_0\times|D_0|+w_1\times|D_1|+w_2\times|D_2|)=0.13$,因此是非频繁的.

基于强前件类相关规则的定义,可得 6 条规则:

1) $(a_1,d_4)\Rightarrow Y|(ass=0.30\wedge\rho=-0.48)$;
2) $(a_1,d_4)\Rightarrow Z|(ass=0.30\wedge\rho=0.87)$;
3) $(b_2,d_4)\Rightarrow X|(ass=0.32\wedge\rho=-0.57)$;
4) $(b_2,d_4)\Rightarrow Y|(ass=0.32\wedge\rho=0.34)$;
5) $d_4\Rightarrow X|(ass=0.48\wedge\rho=-0.58)$;
6) $d_4\Rightarrow Z|(ass=0.48\wedge\rho=0.58)$.

同上,可发现包含 c_1 但不包含 d_4 的前件类关联规则:

7) $(a_1,c_1)\Rightarrow X|(ass=0.35\wedge\rho=0.42)$;

8) $(a_1,c_1)\Rightarrow Y|(ass=0.35\wedge\rho=-0.54)$;

发现包含 b_2 但不包含 c_1 和 d_4 的前件类关联规则:

9) $b_2\Rightarrow X|(ass=0.59\wedge\rho=-0.54)$;

10) $b_2\Rightarrow Y|(ass=0.59\wedge\rho=0.61)$;

发现仅包含 a_1 的前件类关联规则:

11) $a_1\Rightarrow X|(ass=0.65\wedge\rho=0.61)$;

12) $a_1\Rightarrow Y|(ass=0.65\wedge\rho=-1.00)$;

13) $a_1\Rightarrow Z|(ass=0.65\wedge\rho=0.42)$.

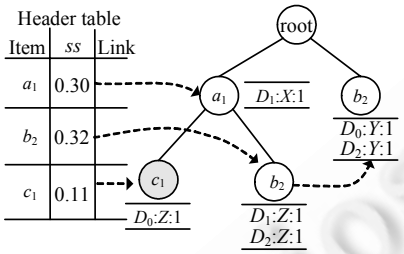


Fig.3 Conditional pattern-base of the item $\langle D,d_4\rangle$

图 3 项$\langle D,d_4\rangle$的条件模式基

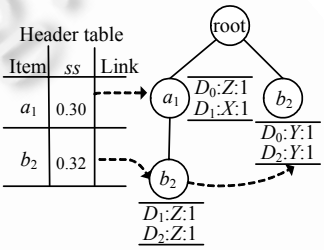


Fig.4 Conditional strong FP-tree of the item $\langle D,d_4\rangle$

图 4 项$\langle D,d_4\rangle$的条件加强 FP-tree

在例 1 中,如果挖掘的是强类关联规则,采用同样的空间支持度阈值 min_ss=0.3,设定较小的空间置信度阈值 min_sc=0.5,则只能生成规则 $b_2\Rightarrow Y|(ss=0.35\wedge sc=0.59)$和 $a_1\Rightarrow X|(ss=0.41\wedge sc=0.63)$.即使类别 Z 出现的频率不是特别少,也挖掘不到任何支持 Z 的规则.如果为此调低支持率的阈值,则势必会导致生成的加强 FP-tree 的增长,从而会降低本文分类器的训练性能,同时增加针对多数类的冗余规则.如果为此调低置信度的阈值,则会产生较多针对多数类的满足置信度同时为负相关的误导性规则,降低分类的准确率.因此,本文采用强前件类相关规则进行分类.

当在加强 FP-tree FPT 中挖掘得到强前件类相关规则 $I\Rightarrow c_i$ 时,则将该规则插入到分类规则树 CR-tree 中,同时触发树中的规则剪枝.剪枝策略是:如果规则 r_1 是 r_2 的泛化规则,同时,$\rho(r_1)>\rho(r_2)>0$ 或者$\rho(r_1)<\rho(r_2)<0$ 或者$(\rho(r_1)=\rho(r_2))\wedge(ass(r_1)>ass(r_2))$时,则将树中的规则 r_2 剪掉.在例 1 中,因为发现规则 11 而剪掉规则 7,发现规则 12 而剪掉规则 1 和规则 8,发现规则 10 而剪掉规则 4,发现规则 5 而剪掉规则 3.经上述处理,得到例 1 所对应的 CR-tree 如图 5 所示.

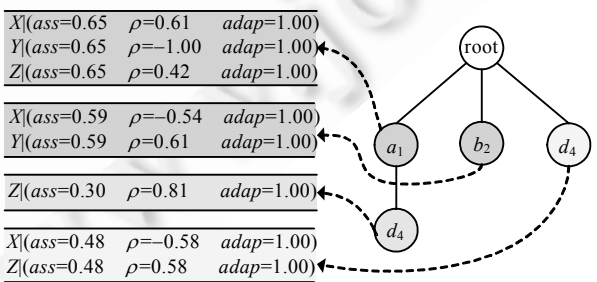


Fig.5 Corresponding classification rule tree CR-tree of D_0

图 5 D_0 所对应的分类规则树 CR-tree

在上图中,$adap$ 是规则对新分类数据集的适应因子,初始化值为 1,将在下节中详细描述其作用和计算方法.下面就基于分布式加强 FP-tree 的分类规则树的构建算法 CRT-Constructing 描述如下:

算法 2. CRT-Constructing.

输入:本地数据集 D_0 所对应的分布式加强 FP-tree *FPT*;

各点数集分类重要度权值$\{w_0,w_1,\ldots,w_{n-1}\}$;

最小空间支持度阈值 min_*ss*;

最小相关系数阈值 min_ρ.

输出:本地数据集 D_0 所对应的分类规则树 *CRT*.

```
CRT_Growth(FPT,null);                          //通过调用该函数完成算法
Procedure. CRT_Growth(Tree,α)
IF (OnlyOnePath(Tree)) {                       //如果 Tree 仅含有单个路径
  P=SinglePath(Tree);                          //P 为 Tree 所包含的单一路径
  FOR (∀β⊆P) {                                 //对于 P 中节点的每个组合β
    IF (!MaxItemsWithSameSS(β)) CONTINUE;      //当存在β_i 使得(β⊆β_i⊆P)∧(ss(β)==ss(β_i))时,β不作处理
    γ=α∪β;                                     //产生模式γ
    IF (ss(γ)≥min_ss)
    FOR (∀c_i∈C) {
      IF (|ρ(γ⇒c_i)|≥min_ρ))                   //如果规则γ⇒c_i 满足最小相关系数阈值
      Insert(γ⇒c_i,CRT);                       //将规则γ⇒c_i 插入到树 CRT 中,同时触发剪枝
    }
  }
}
ELSE {
  FOR (∀f_i⊆L_Tree) {                          //自底向上处理 Tree 项集表中的每一项 f_i
    γ=α∪f_i;
    IF (ass(γ) ≥min_ss)
    FOR (∀c_i∈C) {
      IF (|ρ(γ⇒c_i)|≥min_ρ))
      Insert(γ⇒c_i,CRT);
    }
    Tree_γ=GenerateConditionFPT(γ);            //产生γ的条件模式树 Tree_γ
    IF(Tree_γ≠∅)
      CRT_Growth(Tree_γ,γ);
  }
}
```

算法 CRT-Constructing 以经典算法 FP-growth 为原型,因在加强 FP-tree 中保存了类标信息,所以将强前件类相关规则的挖掘和频繁模式的发现合并为一步,并将发现长频繁模式的问题转换成递归地发现一些短模式,然后连接后缀.它使用最不频繁的项作后缀,提供了较好的选择性.该方法只需在第 1 步生成的加强 FP-tree 中递归挖掘,大大降低了搜索开销,有着较高的运行性能.

3 大数据环境中辅助关联分类器的动态调整

在空间数据集 *D* 上为本地数据集 D_0 构建完分类规则树 *CRT* 后,即可使用 *CRT* 为 D_0 新采集数据元组进行分类.现有的分类器为静态分类器,即,在下次重新训练前分类器在分类应用过程中保持不变.然而相对于原训练样本集,新采集的数据可能存在渐变的数据分布特征和类别迁移现象.这在一定程度上影响了分类准确率,同

时加速了分类器的退化和重新训练的频率.为使分类器能够适应新的分类数据,提高分类准确率,同时充分利用领域专家反馈的实时分类结果,提出一种基于适应因子的分类器动态调整方法.

定义 6(适应因子(adaptive factor,简称 adap)). 设 $I\Rightarrow c_i$ 是本地数据集 D_0 所对应的在空间数据集 D 上挖掘得到的一条类关联规则,在为 D_0 所新采集数据进行分类过程中,共有 n 个元组匹配该规则,且根据领域专家的实时分类结果反馈,共有 n_1 个元组的分类结果正确,$n-n_1$ 个元组的分类结果错误.则定义规则 $I\Rightarrow c_i$ 对新分类数据的适应因子为

$$adap(I\Rightarrow c_i)=\frac{1}{1-\log_e n_1/n} \tag{4}$$

由公式(4)可知,规则 $I\Rightarrow c_i$ 的适应因子 $adap(I\Rightarrow c_i)$是关于该规则对新数据分类正确率 n_1/n 的递增函数,且当 $n_1/n=0$ 时,$adap(I\Rightarrow c_i)=0$;当 $n_1/n=1$ 时,$adap(I\Rightarrow c_i)=1$.适应因子度量了规则对新分类数据的适应程度,该值越大,说明该规则对新分类数据适应度越高,在分类过程中所起的作用就越大.

在为本地数据集 D_0 构建的分类规则树 CRT 中,每条规则的适应因子初始化值为 1.在使用分类器 CRT 为 D_0 新采集的元组 t 进行分类时,可计算元组 t 对各个类别 c_i 的归属度,归属度最大的类别,即为 t 的分类标号.

定义 7(归属度(belonging degree,简称 bel)). 给定本地数据集 D_0 的分类规则树 CRT 和 D_0 上一个待分类元组 t,假设 CRT 中 t 所匹配的类别为 c_i 的强前件类相关规则集为$\{I_1\Rightarrow c_i,I_2\Rightarrow c_i,\ldots,I_k\Rightarrow c_i\}$,则元组 t 对类别 c_i 的归属度为

$$bel(t,c_i)=\begin{cases}ss(c_i)\times\sum_{j=1}^{k}adap(I_j\Rightarrow c_i)\times\rho(I_j\Rightarrow c_i), & k\geqslant 1\\ 0, & k=0\end{cases} \tag{5}$$

在上式中,$adap(I_j\Rightarrow c_i)$为规则 $I_j\Rightarrow c_i$ 对新分类数据集的适应因子;$\rho(I_j\Rightarrow c_i)$为规则前件 I_j 的出现对后件 c_i 的出现所起的推进或抑制作用的度量,即为该规则的相关系数;$ss(c_i)$是类别 c_i 本身出现的加权概率,即为 c_i 的空间支持度.元组 t 对类别 c_i 的归属度是在 t 所包含前件发生的状态下,对类别 c_i 可能发生几率的度量,其中考虑了规则对新分类数据集的适应状态以及规则前件对类别出现所起的抑制作用等多种因素.

分类时,计算待分类元组 t 对各类别 $c_i(0\leqslant i\leqslant m)$的归属度.最终,$t$ 所属的类别 c_i 满足:$\neg\exists c_j|bel(t,c_j)>bel(t,c_i)$,即,$t$ 属于其归属度最大的那个类别.基于分类规则树对新采集样本进行分类的算法较为简单,这里不再详述.需要指出的是,虽然分类器 CRT 在分类应用过程中能够动态调整,但也会随着新数据集的不断增加而降低分类准确率;当准确率下降超过阈值时,需要对其进行重新训练.

4 实 验

为了验证本文大数据环境中辅助关联分类算法 AAC 的有效性,将分别从分类器训练性能、分类准确率和分类器面向动态数据集的适应能力这 3 个方面进行分析.

4.1 算法AAC的训练性能分析

实验首先分析本文辅助关联分类器 AAC 的训练性能.为了验证本文方法的运行效率,将与经典分布式关联挖掘算法 FMD[16]、较经典的算法 FMAGF[20]以及较新算法 DMARF[22]进行性能对比分析.

实验数据:该项实验采用来源于科技部“十五”攻关课题的中医小儿肺炎病例数据集 Pneumonia,该数据集的原始信息见表 5.

该数据集共含有 76 维标称型属性或序数型属性.为了有效地对算法 AAC 的训练性能进行验证,本文还采用了基于遗传算法的数据生成技术,等比例地扩充每个中心的数据,含已采集病例共生成约 1 500K 条分类数据.

实验环境:采用 7 台机器对应存放各中心的数据,有 4 台机器位于 100M 校园局域网环境中,有 3 台机器通过 10M 宽带接入广域网环境中.每台机器的配置均为:CPU:i5-2400,3.1GHz;内存:8G,操作系统:Win7 Ultimate.

实验内容:(1) 采用 1 000K 数据,选“SY”为本地数据集,图 6 列出了 AAC 算法与 FMD 和 DMARF 算法的执行时间随支持度阈值变化的结果;(2) 采用支持度阈值为 0.02,图 7 给出了 AAC 算法与 FMD 和 DMARF 算法的

执行时间随数据量变化的结果.

Table 5 Information of Pneumonia dataset

表 5 Pneumonia 数据集信息描述

中心	辽宁沈阳	辽宁大连	黑龙江	山东	上海	四川	广东	总计
代码	SY	DL	HLJ	SD	SH	SC	GD	
类别 1:风寒闭肺	0	1	1	0	3	25	10	40
类别 2:风热闭肺	16	861	206	763	87	52	210	2 195
类别 3:痰热闭肺	1394	169	711	25	659	726	772	4 456
类别 4:湿热闭肺	1	26	0	0	212	33	227	499
类别 5:阴虚闭肺	91	57	0	12	29	51	114	354
类别 6:肺脾气虚	3	74	31	238	22	382	80	830
类别 7:其他	6	23	127	216	9	0	152	533
总计	1 511	1 211	1 076	1 254	1 021	1 269	1 565	8 907

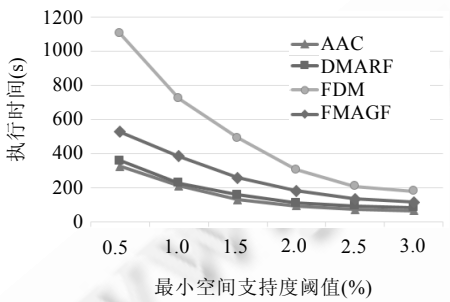


Fig.6 Execution time along with different min_*supports*

图 6 执行时间随最小支持度的变化

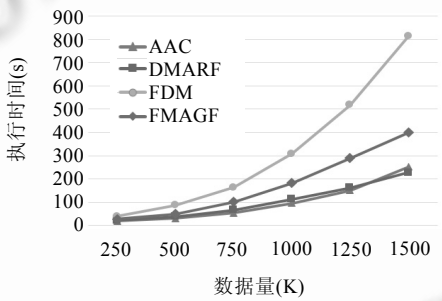


Fig.7 Execution time along with different data amounts

图 7 执行时间随数据量的变化

由图 6 和图 7 可以看出:要生成用于分类的全局频繁项集,与基于 FP-growth 思想的 FMAGF 算法、DMARF 算法以及本文的 AAC 算法相比,经典算法 FMD 随着最小支持度阈值的减小或数据量的增加,其执行时间都有较大的延长趋势;而在基于 FP-growth 思想的算法中,DMARF 的执行性能明显要优于 FMAGF,本文算法 AAC 的执行性能又略优于 DMARF.这说明本文关联分类器训练算法 AAC 具有较好的伸缩性.另外,以算法 DMARF 的思路,先汇总挖掘全局频繁项集的超集,再分布式地求解剪枝,也是一种可行的方案.

4.2 算法AAC的分类准确率分析

接下来对本文分类器 AAC 的分类准确率进行分析.主要从以下 3 个方面进行测试:一是针对普通单分类数据集,测试本文分类器 AAC 与经典关联分类器 CMAR[7]的分类准确率;二是针对不平衡单数据集,测试本文分类器 AAC 与不平衡数据分类器 PCBoost[25]的分类准确率;三是针对分布式分类数据集,测试本文分类器 AAC 在有无横向加权方式下的分类准确率.

首先,针对第 1 项测试,选择了 UCI 机器学习库中的 10 个数据集,采用 10-折交叉验证方法,即,把所有样本分成 10 等份,每次将其中的 9 份作为训练集,剩下的 1 份作为测试集,计算测试集的分类准确率,将 10 次准确率的平均值作为该数据集的准确率.表 6 分别列出了各数据集特征以及本文分类算法 AAC 和经典关联分类算法 CMAR 在各数据集上的分类准确率.

由表 6 可以看出,本文分类算法 AAC 的分类准确率仅在 Hypo 和 Zoo 两个数据集上略差于经典关联分类算法 CMAR;而在不平衡数据集 Glass 和 Vehicle 上,ACC 的分类准确率都比 CMAR 有明显的提升.总体上,AAC 的分类准确率要优于 CMAR.

其次,针对第 2 项测试,为评估算法的性能,采用文献[25]中所选择的 10 个 UCI 数据集和对应的类别,在评价时,选择机器学习领域对于不平衡数据分类最常用的评价标准——几何平均准则(*g*-mean).它反映了分类算法

对不平衡数据集两类样本分类性能的均衡程度,故能较全面地反映不平衡数据集分类算法的性能.表 7 列出了各数据集特征以及本文分类算法 AAC 和较新的不平衡数据分类算法 PCBoost 在各数据集上的 *g*-mean 值.

Table 6 Accuracy comparison of CMAR and AAC on UCI datasets

表 6 AAC 和 CMAR 算法在 UCI 数据集上的准确率比较

数据集	实例数	属性数	类别数	CMAR (%)	AAC (%)
Austral	690	14	2	86.1	87.3
Breast	699	10	2	96.4	96.6
Cleve	303	13	2	82.2	83.7
Glass	214	9	7	70.1	78.7
Heart	270	13	2	82.2	84.8
Hypo	3 163	25	2	98.4	97.1
Iris	150	4	3	94.0	94.5
Vehicle	846	18	4	68.8	76.4
Waveform	5 000	21	3	83.2	83.3
Zoo	101	16	7	97.1	96.8
Average	–	–	–	85.9	87.9

Table 7 *g*-mean value comparison of AAC and PCBoost on imbalanced datasets

表 7 AAC 和 PCBoost 算法面向不平衡数据集的 *g*-mean 值比较

数据集	实例数	少数类	多数类	类分布	PCBoost (%)	AAC (%)
Sonar	208	97	111	0.47:0.53	88.9	91.8
Monk2	169	64	105	0.37:0.63	61.7	60.9
Ionosphere	351	126	225	0.35:0.65	89.4	92.5
Breast	699	241	458	0.34:0.66	98.7	96.3
Vehicle	846	199	647	0.23:0.77	96.2	95.4
Segment	2 310	330	1 980	0.14:0.86	99.5	97.2
Glass	214	29	185	0.13:0.87	94.9	93.1
Satimage	6 435	626	5 809	0.10:0.90	82.2	83.7
Vowel	990	90	900	0.09:0.91	92.1	94.4
Abalone	731	42	689	0.06:0.94	70.4	65.6
Average	–	–	–	–	87.4	87.1

由表 7 可以看出,本文分类算法 AAC 的 *g*-mean 值在 4 个数据集上要优于 PCBoost 算法,在 6 个数据集上要差于 PCBoost 算法,平均值也略差于 PCBoost 算法.虽然本文分类算法面对不平衡数据集的适应能力略差于 PCBoost 算法,但在训练分类器时不需要做数据采样、弱分类器合并等额外工作,因此也是一种有效的不平衡数据分类方案.

再次,针对第 3 项测试,采用原始采集的中医小儿肺炎数据集,依次选择各中心为本地数据集.算法 AAC 分别以仅采用本地数据集(Local-AAC)、平等地采用全局数据集(Global-AAC)和以横向加权的方式采用全局数据集(Weighted-AAC)这 3 种方式进行测试.Local-AAC 进行测试时以本中心 1/2 的数据用于训练,1/2 的数据用于分类测试.在 Global-AAC 和 Weighted-AAC 进行测试时,以本中心 1/2 的数据和其他中心的所有数据用于训练,本中心 1/2 的数据用于测试.图 8 列出了上述 3 种方式在各中心上的分类准确率.

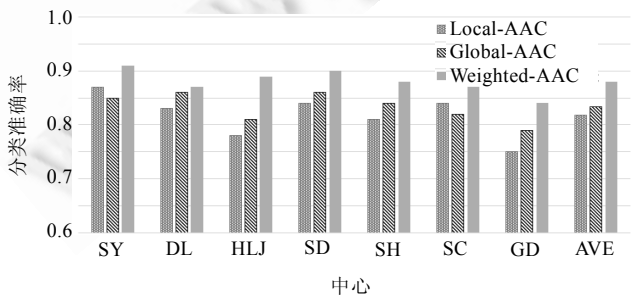


Fig.8 Classification accuracy of the algorithm AAC on each center

图 8 算法 AAC 在各中心上的分类准确率

在图 8 中,AVE 代表各中心的平均分类准确率.由图 8 可以看出,使用全局数据集在各中心上的总体分类准确率要略高于仅采用本地数据集的分类准确率,但以横向加权方式训练出的分类器 Weighted-AAC 的分类准确率要明显优于 Local-AAC 和 Global-AAC.这说明本文横向加权的方式可以有效地利用全局数据集,提高本地数据的分类准确率.

4.3 基于专家反馈结果的分类器AAC动态调整的性能分析

实验最后,对本文分类算法 AAC 基于专家反馈结果进行动态调整前后的分类准确率进行分析.采用原始中医小儿肺炎数据集 Pneumonia 和 UCI 中含有数据量较大的 Waveform 数据集.对 Pneumonia 数据集进行测试时,开始以 1~2 月份的数据作为训练集,生成的分类器依次对 3~4 月份、5~6 月份、7~8 月份、9~10 月份和 11~12 月份的数据进行分类准确率测试.图 9 给出了不进行动态调整和基于专家实时反馈结果进行动态调整的 AAC 分类器的分类结果.对 Waveform 数据集进行 2 折递进测试法,即,将该数据集随机分成 5 等份,选其中一份作为训练集,然而依次选择其他的 4 份为测试集.图 10 给出了 AAC 算法分别在有无动态调整状态下的分类结果.

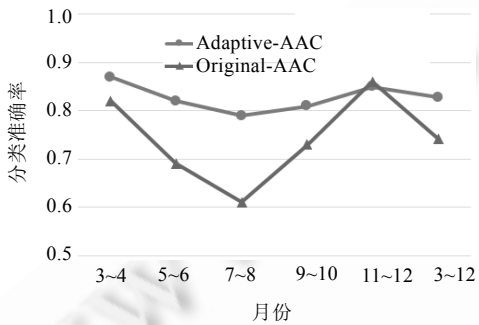


Fig.9 Step-Up accuracy on Pneumonia dataset

图 9 Pneumonia 数据集上的递进分类准确率

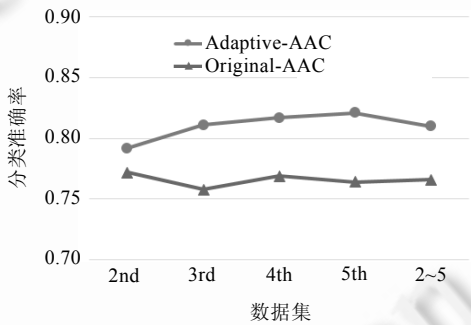


Fig.10 Step-Up accuracy on Waveform dataset

图 10 Waveform 数据集上的递进分类准确率

在图 9 和图 10 中,Original-AAC 表示分类器 AAC 在分类过程中不进行动态调整,而 Adaptive-AAC 是指在分类过程中基于专家反馈结果进行动态调整的分类器 AAC.从图 9 可以看出,无动态调整的分类器 Original-AAC 随着病例数据采集时间的变化,其分类准确率的变化非常明显;而进行动态调整后的分类器 Adaptive-AAC 相对变化幅度较小.由图 10 可以看出,对于随机等分的 Waveform 数据集,Original-AAC 和 Adaptive-AAC 随着分类数据集的变化,它们的分类准确率变化都不大;但无论是针对哪个数据集,Adaptive-AAC 的分类准确率都要较明显地优于 Original-AAC.这说明本文提出的基于专家反馈结果的分类器动态调整算法可以增强分类器面向动态数据集的分类性能.

由以上实验分析可知,本文提出的大数据环境中分布式辅助关联分类算法 AAC 具有较高的分类器训练性能和分类准确率,并且能够针对辅助分类应用,提升分类器面向动态数据集的分类性能,是一种有效的大数据环境中辅助分类应用方法.

5 结 论

随着数据中所蕴含的价值越发被人们重视以及数据应用的不断延伸,大数据环境中辅助分类必然会成为分类应用领域的一个重要分支.然而,现有的分类研究仍缺乏对此类应用的关注.为此,本文提出了一种考虑分布式数据集在空间和时间上类别分布会发生变化的关联分类算法.该算法首先通过横向加权考虑空间各点数据集对本地分类器训练的重要度,构建一棵全局加权 FP-tree;再在其上以“前件空间支持度-相关系数”为度量框架,挖掘产生强前件相关规则集,并以树的形式组织成本地分类器;接下来在分类器应用过程中,根据专家实时反馈的结果对其进行动态调整.实验结果表明,本文算法具有较高的分类器训练性能和分类准确率;同时,面向类别分布不平衡的数据集和动态变化的数据集均有较高的适应能力,是大数据环境中的一种有效的辅助分类

方法.

本文首先分析了大数据环境下辅助分类所面临的问题,接下来描述了本文大数据环境中分布式辅助分类模型,然后给出了模型中的两个核心内容——大数据环境中关联分类器构建算法和辅助分类器动态调整算法,最后给出了针对本文方法的实验分析数据.文中方法需设定的阈值“空间支持度”对算法的影响较大:如果设置过小,则会导致所产生的需要在各点上传输的 FP-tree 过大,从而影响分类器的训练性能;如果设置过大,则又会降低分类器的准确率.通过人工尝试的方式会浪费一定的时间,下一步我们将会探索该阈值的自动确定方法.另外,还会尝试采用其他分类方法进行大数据环境下的辅助分类应用.

References:

[1] Jan TK, Wang DW, Lin CH, Lin HT. A simple methodology for soft cost-sensitive classification. In: Proc. of the ACM SIGKDD Conf. on Knowledge Discovery and Data Mining. New York: ACM Press, 2012. 141−149. [doi: 10.1145/2339530.2339555]

[2] Wu XD, Zhu XQ, Wu GQ, Ding W. Data mining with big data. IEEE Trans. on Knowledge and Data Engineering, 2014,26(1): 97−107. [doi: 10.1109/TKDE.2013.109]

[3] Wang XF, Dong D, Liang MX, Zhang B, Zhang MW. Thinking and methods concerning applying data-mining technique in clinical efficacy evaluation of TCM. Chinese Journal of Integrated Traditional and Western Medicine, 2007,27(10):949−951 (in Chinese with English abstract). [doi: 10.3321/j.issn:1003-5370.2007.10.031]

[4] Helmbold DP, Long PM. Tracking drifting concepts by minimizing disagreements. Machine Learning, 1994,14(1):27−45. [doi: 10.1007/BF00993161]

[5] Agrawal R, Srikant R. Fast algorithms for mining association rules. In: Proc. of the Int’l Conf. on Very Large Data Bases (VLDB). San Fransrsco: Morgan Kaufmann Publishers, 1994. 487−499.

[6] Liu B, Hsu W, Ma Y. Integrating classification and association rule mining. In: Proc. of the 4th Int’l Conf. on Knowledge Discovery and Data Mining. New York: ACM Press, 1998. 80−86.

[7] Li WM, Han JW, Pei J. CMAR: Accurate and efficient classification based on multiple class-association rules. In: Proc. of the IEEE Int’l Conf. on Data Mining. Piscataway: IEEE, 2001. 369−376. [doi: 10.1109/ICDM.2001.989541]

[8] Yin XX, Han JW. CPAR: Classification based on predictive association rules. In: Proc. of the SIAM Int’l Conf. on Data Mining. Philadelphia: SIAM, 2003. 331−335. [doi: 10.1137/1.9781611972733.40]

[9] Dong G, Zhang X, Wong L, Li J. CAEP: Classification by aggregating emerging patterns. In: Proc. of the 2nd Int’l Conf. of Discovery Science. Berlin: SpringerVerlag, 1999. 30−42. [doi: 10.1007/3-540-46846-3_4]

[10] Wang J, Karypis G. HARMONY: Efficiently mining the best rules for classification. In: Proc. of the SIAM Int’l Conf. on Data Mining. Philadelphia: SIAM, 2005. 205−216. [doi: 10.1137/1.9781611972757.19]

[11] Shafer JC, Agrawal R, Mehta M. SPRINT: A scalable parallel classifier for data mining. In: Proc. of the 22nd Int’l Conf. on Very Large Data Bases. San Fransrsco: Morgan Kaufmann Publishers, 1996. 544−555.

[12] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Machine Learning, 1997,29(1):131−163. [doi: 10.1023/A:1007465528199]

[13] Song HH, Lee SW. A self-organizing neural tree for large-set pattern classification. IEEE Trans. on Neural Networks, 1998,9(5): 369−380. [doi: 10.1109/72.668880]

[14] Ghosh AK, Chaudhuri P, Murthy CA. On visualization and aggregation of nearest neighbor classifiers. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2005,27(10):1592−1602. [doi: 10.1109/TPAMI.2005.204]

[15] Agrawal R, Shafer J. Parallel mining of association rules. IEEE Trans. on Knowledge and Data Engineering, 1996,8(6):962−969. [doi: 10.1109/69.553164]

[16] Cheung DW, Han JW, Ng VT, Fu AW, Fu Y. A fast distributed algorithm for mining association rules. In: Proc. of the IEEE 4th Int’l Conf. on Parallel and Distributed Information Systems. Miami Beach: IEEE Press, 1996. 31−44. [doi: 10.1109/PDIS.1996.568665]

[17] Schuster A, Wolff R. Communication efficient distributed mining of association rules. In: Proc. of the ACM SIGMOD Int’l Conf. on Management of Data. New York: ACM Press, 2001. 473−484. [doi: 10.1145/375663.375728]

[18] Han J, Pei J, Yin Y. Mining frequent patterns without candidate generation. In: Proc. of the ACM Conf. on Management of Data (SIGMOD). New York: ACM Press, 2000. 1−12. [doi: 10.1145/342009.335372]

[19] Zaiane OR, EI-Hajj M, Lu P. Fast parallel association rule mining without candidacy generation. In: Proc. of the IEEE Int'l Conf. on Data Mining. Washington: IEEE Computer Society Press, 2001. 665−668. [doi: 10.1109/ICDM.2001.989600]

[20] Yang M, Sun ZH, Ji GL. Fast mining of global frequent itemsets. Journal of Computer of Research and Development, 2003,40(4): 620−626 (in Chinese with English abstract).

[21] Song BL, Qin Z. Fast mining algorithm for distributed global frequent itemset. Journal of Xi'an Jiaotong University, 2006,40(8): 923−927 (in Chinese with English abstract). [doi: 10.3321/j.issn:0253-987X.2006.08.013]

[22] He B. Distributed algorithm for mining association rules based on FP-tree. Control and Decision, 2012,27(4):618−622 (in Chinese with English abstract).

[23] Chawla NV, Japkowicz N, Kolcz A. Editorial: Special issue on learning from imbalanced data sets. ACM SIGKDD Explorations Newsletter, 2004,6(1):1−6. [doi: 10.1145/1007730.1007733]

[24] He HB, Garcia EA. Learning from imbalanced data. IEEE Trans. on Knowledge and Data Engineering, 2009,21(9):1263−1284. [doi: 10.1109/TKDE.2008.239]

[25] Li XF, Li J, Dong YF, Qu CW. A new learning algorithm for imbalanced data—PCBoost. Chinese Journal of Computers, 2012, 35(2):202−209 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2012.00202]

附中文参考文献:

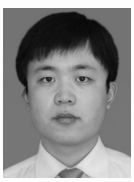
[3] 王雪峰,董丹,梁茂新,张斌,张明卫.数据挖掘技术在小儿肺炎中医临床疗效评价研究中应用的思路与方法.中国中西医结合杂志,2007,27(10):949−951. [doi: 10.3321/j.issn:1003-5370.2007.10.031]

[20] 杨明,孙志挥,吉根林.快速挖掘全局频繁项目集.计算机研究与发展,2003,40(4):620−626.

[21] 宋宝莉,覃征.分布式全局频繁项目集的快速挖掘方法.西安交通大学学报,2006,40(8):923−927. [doi: 10.3321/j.issn:0253-987X.2006.08.013]

[22] 何波.基于频繁模式树的分布式关联规则挖掘算法.控制与决策,2012,27(4):618−622.

[25] 李雄飞,李军,董元方,屈成伟.一种新的不平衡数据学习算法 PCBoost.计算机学报,2012,35(2):202−209. [doi: 10.3724/SP.J.1016.2012.00202]



张明卫(1979－),男,山东胶州人,博士,讲师,CCF 会员,主要研究领域为服务计算,数据挖掘.

刘莹(1981－),女,博士,讲师,CCF 会员,主要研究领域为服务计算.

朱志良(1962－),男,博士,教授,博士生导师,CCF 会员,主要研究领域为云计算,复杂网络.

张斌(1964－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为服务计算,信息集成.