doi:10.3969/j.issn.1001-4616.2019.04.008

数据流决策树分类方法综述

贾　涛,韩　萌,王少峰,杜诗语,申明尧

(北方民族大学计算机科学与工程学院,宁夏 银川 750021)

[**摘要**]　数据流的特征是海量的、高速流动的、实时处理的. 由于一些数据分布随着时间而改变,因此将这些数据流称为概念漂移. 首先按照分类模型对数据流决策树进行分类,分为单分类决策树和集成分类决策树. 单分类模型分为快速决策树、变异决策树和其他决策树算法. 集成分类模型分为衍生快速决策树和随机决策树变体算法. 其次介绍了概念漂移处理技术,包括概念漂移问题的描述、常见的概念漂移处理技术和用于解决概念漂移的决策树算法.接着介绍了增量模型决策树算法,最后对本文介绍的决策树算法进行分析总结.

[**关键词**]　数据流挖掘,分类,决策树,概念漂移,集成分类

[**中图分类号**]TP3　[**文献标志码**]A　[**文章编号**]1001-4616(2019)04-0049-12

Survey of Decision Tree Classification Methods over Data Streams

Jia Tao,Han Meng,Wang Shaofeng,Du Shiyu,Shen Mingyao

(School of Computer Science and Engineering,North Minzu University,Yinchuan 750021,China)

Abstract: The data streams are characterized by massive, high-speed and real-time processing. Since some data distributions change over time,these data streams are called concept drift. Firstly,the data streams decision trees are classified according to the classification models,which are divided into single classification decision trees and ensemble classification decision trees. Single classification models are divided into very fast decision tree,mutation decision trees and other decision tree algorithms. The ensemble classification models are divided into derivative very fast decision tree and random decision tree variant algorithms. Secondly,the concept drift processing technologies are introduced,including the description of the concept drift problem,the common concept drift processing technology and the decision tree algorithm for solving the concept drift. Then the incremental model decision tree algorithm is introduced. Finally,the decision tree algorithms introduced in this paper is analyzed and summarized.

Key words:data streams mining,classification,decision tree,concept drift,ensemble classification

数据流模型广泛应用于社会生产和生活的各个领域,它是未来数据发展的一个主要趋势. 它也已经成为当前的一个研究热点. 常见的数据流分类方法包括神经网络[1-2]、关联/分类规则、支持向量机[3-4]、决策树[5-6]等. 数据流决策树是最有效的分类方法之一. 决策树算法已经成功地应用于许多应用领域的分类,在医疗诊断、天气预报、金融分析、顾客分类、身份识别、网络安全和行为分析等领域逐渐发挥着越来越重要的作用. 与此同时,国内外研究工作者也在不断丰富数据流决策树算法的理论知识. 数据流与传统的数据集不同,具有动态、无限、高维、有序、高速和变化等特性[7]. 不同于数据仓库,它是实时产生,一般不被存储,而且不一定服从同一分布,这样大量的数据蕴含着当前有效的信息,所以要求快速地处理,尽量短时间内挖掘出其中有用的信息. 鉴于数据流的高速性和连续性,数据流算法应是动态增量的,亦必须是高效的. 传统的数据挖掘技术已不再适合数据流环境. 因此,数据流环境下的数据挖掘研究将面临更大的机遇和挑战性[8].

近年来,国内外研究者在数据流分类方面做了大量研究. 数据流决策树分类算法按照分类模型分为单分类决策树和集成分类决策树. 单分类决策树算法包括快速决策树(very fast decision tree,VFDT)[9]、概

收稿日期:2019-06-17.

基金项目:国家自然科学基金项目(61563001).

通讯联系人:韩萌,博士,副教授,研究方向:数据挖掘. E-mail:861254268@qq.com,2003051@nun.edu.cn

念漂移快速决策树(conception-adapting very fast decision tree,CVFDT)[10]、高斯决策树(Gaussian decision tree,GDT)[11-14]、决策树进化算法(evolutionary algorithm for decision tree,EVO-Tree)[15-16]、概念自适应决策树进化(concept-adapting evolutionary algorithm for decision tree,CEVOT)算法[17]、进化模糊最小极大决策树(evolving fuzzy min-max decision tree,EFMMDT)[18]、基于分数近似的决策树(decision trees based on the fractions approximation,DTFA)算法[19]、隐私保护快速决策树(privacy preserving fast decision tree,PPFDT)[20]和极速决策树(extremely fast decision tree,EFDT)算法[21]等. 集成分类决策树算法包括霍夫丁选项树(Hoeffding option tree,HOT)[22]、不确定处理概念漂移快速决策树(uncertainty-handle and conception-adapting very fast decision tree,UCVFDT)[23]、成本敏感的感知器决策树(CSPT)[24]、增量优化快速决策树(incrementally optimized very fast decision tree,iOVFDT)[25-26]、基于概念漂移数据流的集成决策树(ensemble decision trees for concept-drifting data streams,EDTC)[27]、随机决策树(random decision tree,RDT)[23]和基于随机决策树的数据流分类算法等. 用于检测和处理概念漂移的决策树方法有概念漂移快速决策树(CVFDT)、不确定处理概念漂移快速决策树(UCVFDT)[28]、概念漂移随机决策树(concept drifts in random decision tree,CDRDT)算法[29]、自适应分级滑动窗口决策树算法(adapting grading slide-window decision tree,AGSW-DT)[30]、自适应快速决策树(AFDT)[31]、基于概念漂移数据流的集成决策树(EDTC)和成本敏感的感知器决策树(CSPT)等.

本文主要是对现有的数据流决策树分类算法进行分析总结. 第 1 章主要将决策树算法分为单分类模型和集成分类模型进行阐述. 第 2 章首先介绍了概念漂移处理技术,包括概念漂移问题的描述、常见的漂移处理技术和用于解决概念漂移的决策树算法;其次介绍了增量模型决策树算法. 第 3 章对本文中的数据流决策树分类算法进行分析与总结. 文章最后提出了数据流分类现阶段存在的问题.

1 决策树分类方法

数据流决策树分类模型主要分为两类,即单分类决策树模型和集成分类决策树模型. 其中,单分类模型技术维护和增量更新单个分类模型,它能有效地对概念漂移做出回应. 相对于单个模型,集成模型需要比单分类更简单的技术更新模型,且同样可以有效地处理概念漂移. 集成分类器处理概念漂移问题时优于单个分类器[8].

1.1 单分类决策树模型

单分类器模型是不断用新的数据来递归地更新自身结构,使自身结构能够适应流中数据的变化,并在流中对实例能够准确分类. 最早提出用于处理数据流的决策树分类算法是基于 Hoeffding 树,后继很多决策树算法也是基于 Hoeffding 不等式设计而来的. 因此对 Hoeffding 算法作以下介绍.

Hoeffding 树算法的一个关键特性是,它可以保证产生的树渐近地接近批量学习分类器生成的树. 换句话说,Hoeffding 树算法的增量特性不会显著影响其生成树的质量. 为了做到这一点,需要定义两个决策树之间不一致的概念,如定义 1 和定义 2. 设 $P(X)$ 是被观察到的属性向量 X 的概率,$I(.)$ 为指标(评估)函数,如果其参数为 true 则返回 1,否则为 0.

定义 1[32-33] 两个决策树 DT_1 和 DT_2 之间的差异 Δ_u 是它们产生不同类预测的概率,如式(1)所示.

$$\Delta_u(DT_1,DT_2)=\sum_X P(X)I[DT_1(X)\neq DT_2(X)]. \tag{1}$$

如果两个内部节点包含不同的测试结果,那么这两个内部节点是不同的. 如果它们包含不同的类预测,那么这两个叶子也是不同的,并且内部节点与叶子是不同的. 另外,如果树中的两条路径长度不同,或者至少在一个节点上不同,那么也要考虑它们是不同的.

定义 2[32-33] 两种决策树 DT_1 和 DT_2 之间的差异率 Δ_i 是指一个示例通过 DT_1 的路径与通过 DT_2 路径不同的概率,如式(2)所示.

$$\Delta_i(DT_1,DT_2)=\sum_X P(X)I[Path_1(X)\neq Path_2(X)], \tag{2}$$

式中,$Path_i(X)$ 是示例 X 到树 DT_i 的路径. 两种决策树 DT_1 和 DT_2 在某种意义上有如下式(3)关系.

$$\forall_{DT_1,DT_2}\Delta_u(DT_1,DT_2)\leqslant\Delta_i(DT_1,DT_2). \tag{3}$$

定理 1[32-33] 如果 HT_Φ 是由 Hoeffding 树算法生成的树,所需的概率为 Φ,给出无穷多个例子,DT_* 是

渐近批处理树,p 是叶节点概率,则有如下关系式(4)成立.

$$E[\Delta_i(HT_\delta, DT_*)] \leqslant \phi/p. \tag{4}$$

Hoeffding 界表明,以概率 $1-\delta$,范围为 R 的随机变量的真实均值不会与 n 次独立观测后的估计均值相差超过:

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}. \tag{5}$$

式中,$R=\log_2 C$,C 是类的数目,δ 是分裂置信度,n 是叶节点数.

Domingos 等人提出了一种称为 Hoeffding 树的增量决策树算法. 无处不在的流媒体数据出现,使得在线机器学习算法非常流行. 在这种情况下 Hoeffding 树代表了最先进的在线分类算法,经常被用作许多在线分类学习算法的基本模型.

VFDT 算法[33]是一种基于 Hoeffding 不等式(如式(5))对数据流模型进行分类的决策树方法. 即使在最坏的情况下 Hoeffding 树也能在一定时间内对一个实例进行学习,而且所构建的决策树分类精度不差于其他决策树算法. 该算法对每一个实例只扫描一次,而且对数据不进行任何保存,所以适合大量数据流的挖掘. 它与经典的诱导树算法类似,主要区别在于分割属性的选择. 在查看所有示例之后,它不是选择最佳属性(根据给定的分割评估函数),而是使用 Hoeffding 不等式来计算所需的示例数,以用户指定的概率选择合适(正确)的分割节点. VFDT 最主要的贡献在于 VFDT 根据充分的统计数据和 Hoeffding 边界(HB)实现了一种启发式的评估函数. VFDT 能够从数据流中构建决策树,这些数据几乎与构建在完全静态的数据池中的数据相当[17].

通过阅读大量数据流相关的文献,以及对数据流决策树分类算法的学习,可以归纳出:其中一部分决策树算法是基于快速决策树算法设计的,还有一部分决策树算法并不是基于快速决策树设计的. 因此,单分类决策树方法按照生成类型,分为快速决策树的衍生算法和其他类型的决策树方法.

1.1.1　快速决策树的衍生算法

首先,Domingos 等人 2000 年提出了快速决策树(VFDT)算法,并首先将决策树学习应用于数据流挖掘[32],解决了构建决策树过程中训练不足的问题. 但是,VFDT 只能处理离散属性,而不是为连续属性设计的. 因此,2003 年 Gama 等人提出了基于 VFDT 算法的 VFDTc,该算法不仅可以处理离散数据,还可以处理连续数据,并且可以在线合并和分类新信息,只需要对数据进行单次扫描,并且扫描每个数据所需的时间都是相同的. 该分类系统最相关的属性也能够获得类似于标准决策树算法的性能,即使对于中等大小的数据流算法性能也是如此[34]. 针对 VFDT 算法不能很好地处理概念漂移,Hulten 等人提出了 CVFDT 算法[35],这个算法在 VFDT 算法的基础上做了一些优化,每当一个新示例到来时,它不需要从头开始学习新的模型;相反,它通过递增与新示例相对应的计数来更新节点上的充分统计信息,并取消与窗口中最老的示例相对应的计数. 然而,Duda 等人提出了基于分数近似的决策树(DTFA)算法,该算法的新颖之处是允许研究者正确地使用 Hoeffding 不等式来获得所需的边界,并且与 GDT 算法性能进行了比较,虽然 GDT 算法是目前用于构建数据流决策树最佳、最合理的算法,但是实验结果表明了 DTFA 算法的优点,获得的分类结果明显优于传统的决策树算法[19].

近 3 年出现的单分类决策树算法包括 PPFDT 和 EFDT 等. 针对当前数据流挖掘应用中的隐私泄露问题,2017 年陈煜等人提出了一种隐私保护快速决策树(PPFDT)[20]算法. 该算法通过采用添加随机噪声的方法对数据加以隐私保护,并使用阀值算法找到扰动数据流的最佳分裂属性和最佳分裂点,从而直接在扰动数据流上构造决策树. 2018 年 Manapragada 等人提出的极速决策树(Extremely Fast Decision Tree,EFDT)[21]算法等同于 Hoeffding 树,只是它使用 Hoeffding 边界来确定最佳属性上拆分的增益是否超过未拆分的增益或当前拆分属性的增益. 实际上,如果一个节点上不存在拆分属性,而不是仅当第一候选拆分属性的性能优于第二好的候选者时才进行拆分. 那么,当来自第一候选拆分的信息增益非零并且具有所需的置信度时,EFDT 将进行拆分. 从而与 Hoeffding Tree 相比,虽然运行时间变长,但准确率升高并且内存明显降低.

1.1.2　其它类型的决策树算法

除了基于快速决策树衍生的单分类决策树算法以外,还有其它类型的单分类决策树模型. 构造决策树的关键是确定划分所考虑节点的最佳属性. 目前提出了几种解决这一问题的方法. 然而,它们要么在数

学上被错误地证明(例如在 Hoeffding 树算法中),要么在运行中花费大量的时间(例如在 McDiarmid 树算法中). 为了解决确定最佳分裂属性的问题,选择应用高斯决策树(GDT)[11]. 该算法在处理时间较短的情况下,可获得较高的分类精度. Jankowski 等人提出了一种利用概念漂移对数据流进行分类的数据挖掘算法,称为决策树进化算法(EVO-Tree)[15]. 该算法决定使用树的质量进行全局度量,即使用树的大小和精度. 这种方法在不降低预测精度的前提下,删除可能产生错误数据结果的分类器,从而降低最终分类器的时间和空间复杂度. 这种方法的新颖之处在于将树学习机制和进化算法(模拟自然生物进化过程的启发式搜索算法)结合在一起. 在 EVO-Tree 中,决策树是增量的,所有信息都存储在树的内部结构中,并且实验表明该算法处理时间更短并且分类准确率更高. Jankowski 提出的概念自适应决策树进化算法(CEVOT)[17]主要是将进化算法用于决策树的增量归纳中. 该算法是批处理算法 EVO-Tree 的扩展版本. 与离线学习相比,CEVOT 从滑动窗口中学习,不需要对漂移的性质或类型进行任何假设,也不需要对新概念的存在或缺乏进行任何假设,并且该算法的预测精度也是相当高的. 在 Mirzamomen 等人 2016 年提出的进化模糊最小极大决策树(EFMMDT)[18]方法中,内部节点包含可训练的分割测试. 与传统的选择单个属性作为分割测试的决策树相比,该方法每个内部节点都包含一个基于多个属性的可训练函数,不仅提供了处理数据流所需的灵活性,而且提高了算法的稳定性. 其中决策树的每个内部节点都包含一个进化的模糊神经网络. EFMMDT 基于多个属性对实例空间进行非线性分割,使得决策树节点更少、层数更浅. 大量的实验表明,与现有的基准数据流决策树学习算法相比,该算法具有更高的精度,特别是在存在概念漂移的情况下分类精度更高.

1.2　集成分类决策树模型

集成分类模型由多个独立的基础分类器组成. 随着数据流的不断到达,对于不同的数据流,适应当前窗口概念的数据量也有所不同. 因此,单一分类器模型具有一定的局限性[11]. 集成学习使得分类器具有更高精度的特性,可以很好地适应概念的变化. 也是目前最有前途的研究方向之一. 集成(也称为多分类器)是一组单独的组件(基)分类器,它们的预测组合在一起来预测新的传入实例. 集成已被证明是提高预测精度或/和将复杂、困难的学习问题分解为更容易子问题的有效方法[36]. 集成分类器的一般算法过程如图 1 所示.

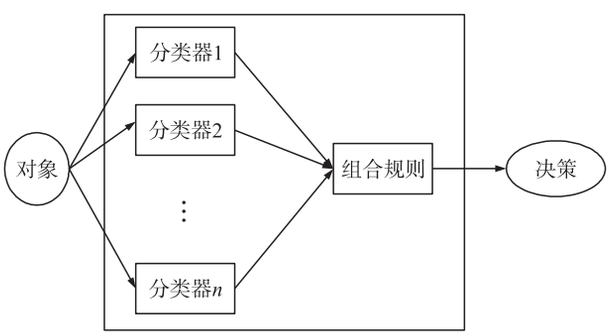


图 1　分类器集成图解

Fig. 1　A diagram of the ensemble classifier

在数据流决策树集成分类算法中,其中一部分集成决策树算法是基于 Hoeffding 不等式设计的,另外一部分是基于随机决策树衍生而来的. 因此,本节将通过以下两个方面对数据流集成决策树算法进行论述.

1.2.1　基于 Hoeffding 不等式的集成分类方法

Pfahringer 等人在 VFDT 的基础上,提出的霍夫丁选项树(HOT)是一个常规的 Hoeffding 树,除了内部决策节点和叶子外,还包含额外的选项节点. 并且允许应用多个测试,从而将多个 Hoeffding 树作为单独的路径. 这个结构使得一个例子可以通过多个不同路径到达多个不同的树节点[22]. 目前对数据流分类的研究主要集中在特定的数据上,通常假定数据流的值是精确的或/和确定的. 然而,由于测量不精确、重复采样和网络误差等原因,具有不确定性的数据在实际应用中是频繁出现的[37]. 在 CVFDT 和 DTU 的基础上,提出了不确定处理概念自适应快速决策树(UCVFDT)算法,该算法既保持了 CVFDT 对概念漂移的高速处理能力,又增加了对不确定性数据的处理能力. 实验研究表明,提出的 UCVFDT 算法能够有效地对具有不确定数值属性的动态数据流进行分类,具有较高的计算效率[30]. Krawczyk 在文[24]构建成本敏感的感知器决策树(CSPT),该算法是在快速感知器决策树(FPDT)[38]的基础之上构建,因为它不但提升了实验运行速度并且提供了高精度的结果,使其非常适合于完成当下的任务. 它的主要优点在于在每片叶子上使用线性感知器. 这样既可以加快决策过程,又可以提高整体准确性[24]. 原始的 VFDT 提出了一种利用 Hoeffding 边界控制树结构节点分裂的简单方法. 它是一种只需要扫描一次数据的数据流处理算法,适用于高速数据流环境. 然而,由于数据流的不完善,会造成过拟合、树尺寸暴增、类分布不平衡等问题,从而

影响分类精度. Yang 等人提出了一种增量优化快速决策树(iOVFDT)算法,用于处理噪声数据流,由于在处理大数据时,很难全面地实现预处理和抽样. 然而该算法在获得较高分类精度的同时可以大大减小树模型尺寸[25]. 基于传统归纳集成的在线分类算法很少关注概念漂移数据流的处理,然而对噪声数据的处理效果较好. 在此基础上,Li 等人提出了一种基于随机决策树的集成算法(EDTC)[27],在虚拟和真实数据流的大量研究中表明,与基于单分类模型和集成模型的几种已知在线算法相比,EDTC 算法性能非常好,并且该方法能有效地解决概念漂移数据流在噪声环境下的学习问题.

1.2.2　随机决策树衍生的集成分类方法

决策树集成分类算法将多个单分类模型组合起来,极大地提高了单分类模型的准确性,然而计算这些联合假设的成本会高很多. 因此 Fan 等人提出随机决策树(RDT)[39]来学习分类器,而它不需要训练集,就能够比单一的最佳假设获得更高的精确性,并且可以与提高或捕获多个最佳假设相媲美. 随机决策树算法,它首先构建了 N 个数据的随机决策树结构. 然后通过逐个扫描训练示例来更新每个节点的统计信息. 对于连续特征,可以离散化,或者随机选择一个分割点(即作为测试条件). 为了说明该算法,使用离散化来描述该算法. 在分类实例 X 时,来自多个随机树的概率输出被平均以估计后验概率[39]. 基于随机决策树的分类算法有随机森林(random forest,RF)[40]、极速决策森林(ultra fast forest tree,UFFT)[41]、数据流随机森林算法(stream random forest,SRF)[42]、随机决策树的挖掘数据流增量算法(semi-random multiple decision tree for data stream,SRMTDS)[43]、动态数据流随机森林算法(dynamic streaming random forests,DSRF)[44]和基于多重半随机概念漂移数据流决策树(multiple semi-random decision trees,MSRT)[45]算法等.

基于随机决策树的数据流分类算法包括随机森林、极速决策森林等. 随机森林(RF)[40]是树预测因子的一个集成,使得每棵树依赖于独立采样的随机向量值,并且森林中所有树的分布全部相同. 森林的泛化错误会随着森林中树木数量的增加而达到极限. 树分类器森林的泛化误差取决于森林中单树的耦合性以及它们之间的相关性. Gama 等人基于随机决策树提出的极速决策森林(UFFT)[41]是一种增量决策树算法,它在处理每个示例时都有一定的时间,并使用 Hoeffding 边界来决定何时在叶子上构建一个分割测试,从而导致决策节点. 根据 Hoeffding 约束条件来评估分割标准的示例数量是合理的. 对于多分类问题,该算法为每一个可能的分裂构建一个二叉树,从而生成一棵森林树. 决策节点和叶子包含了在归纳过程中扮演不同角色的朴素贝叶斯分类器. 利用叶节点中的朴素贝叶斯对测试样本进行分类. 内节点中的朴素贝叶斯具有两种不同的作用. 如果选择了分裂准则,它们可以作为多元分裂测试,并用于检测遍历节点的示例的类分布的变化. 当检测到类分布的变化时,将修剪以该节点为根的所有子树. 在叶子上使用朴素贝叶斯分类器来对测试实例进行分类,使用基于朴素贝叶斯结果的分裂测试,以及在决策节点使用朴素贝叶斯分类器来检测示例分布的变化直接从计算分裂标准所需的足够统计数据中获得,而无需额外的计算. Abdulsalam 等人提出的数据流随机森林算法(SRF)[42]首先生成多个决策树,并基于多个树选取对未标记的记录进行分类. 该算法是标准随机森林算法的扩展,有与标准随机森林算法相当的分类精度,尽管每个数据记录只出现一次. 由于流算法永远不会看到所有数据,所以我们的算法使用节点窗口和树窗口来决定何时开始构建新树,转换边界节点或执行有限形式的剪枝. 这些改进意味着与其他基于流的决策树算法相比,该算法需要更少的标记记录用于训练. 流随机森林算法在许多应用中都能快速处理流. 胡学钢等人提出了一种基于随机决策树的挖掘数据流增量算法(SRMTDS)[43],该算法使用 Hoeffding 不等式,选择最小分裂的例子,使用一个启发式的方法来计算获得数值属性分割阈值的信息增益和一个朴素贝叶斯分类器来估计分类标签的叶子节点. 在处理概念漂移数据流进行分类时,得到了广泛应用. 然而,许多数据流分类算法已经被设计成概念漂移的固定特征,不能处理噪声对概念漂移检测的影响. 2008 年 Abdulsalam 等人考虑数据流的分类问题,引入动态数据流随机森林算法(DSRF)[44],它能够使用基于信息熵的漂移检测技术来处理进化数据流. 该算法处理的是最底层分类边界漂移的多分类问题,并且不会损失实验分类精度. Li 等人提出了基于多重半随机概念漂移数据流决策树(MSRT)[45]算法,将两个滑动窗口进行训练和测试,使用 Hoeffding 不等式确定阈值区分真正的漂移和噪声,并选择分类函数估计概念漂移的错误率. 大量的实验研究表明,MSRT 与 CVFDT 相比,在时间、准确性和健壮性等方面有一定的性能提升.

2　数据流分类关键技术

本章介绍数据流分类关键技术,包括概念漂移处理技术和增量模型处理技术. 处理概念漂移常用的

方法有基于序列的方法(page-hinckley,PH)[46]、累积和(CUSUM)[47]、基于控制图的方法(control charts or statistical process control,SPC)[48]和 ADWIN[49]等. 增量算法是指按照顺序一个接一个(或一批接一批)地处理实例,每次处理一个(一批)实例后更新模型[7].

2.1　概念漂移处理技术

本节首先对概念漂移问题进行描述和分析,然后介绍处理概念漂移的常用技术和用于解决概念漂移的决策树分类方法.

2.1.1　概念漂移问题描述

流挖掘方法被定义为一组用于实时处理数据流的前沿技术,以获取知识. 在分类的特殊情况下,数据流挖掘必须将其行为适应于在虚假的数据分布下的波动,这被称为概念漂移. 概念漂移是指在输入数据的背景下,数据分布的变化. 这里定义了概念漂移:给定预测目标变量 c 和条件变量 X,则一个实例可以表示为(X,c). 首先分类器根据每个类别 $P(c_i)$ 和条件概率 $P(X|c_i)$ 的概率,将数据 X 精确地分配给 c_i,从而实现期望的分类[10]. 使用一个集合 R 用来描述分类器的行为如式(6)所示:

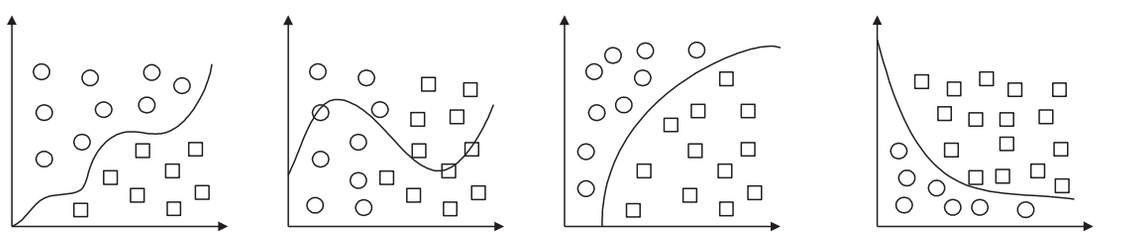
$$R=\{(P(c_1),P(X|c_1)),(P(c_2,P(X|c_2),\cdots,P(c_i),P(X|c_i))\}. \tag{6}$$

给出了每个类的概率及与其相关的条件概率. 根据贝叶斯定理,通过式(7)和 X 属于具有最大概率值的类,得到了属于类 c_i 的观测样本 X 的概率.

$$P(c_i|X)=\frac{P(X|c_i)}{P(X)}, \tag{7}$$

式中,$P(X)$是 X 的一个概率,对于所有的类 c_i 都是常数.

对于预测分类来说,现有的概念漂移问题分为两类,真实概念漂移(real concept drift)和虚假漂移(virtual drift)[24]. 真实概念漂移是指无论 $P(X)$是否发生改变,$P(c|X)$都会发生改变. 虚假漂移是指输入数据改变,即 $P(X)$发生改变,但 $P(c|X)$没有发生改变. 如图 2 所示,图 2(a)是变化前的原始数据,不同形状代表不同的类. 图 2(b)是后验概率变化后的情况,这种情况是实质上的概念漂移. 而图 2(c)是仅有 $P(X)$发生变化的情况,这种情况是虚假漂移. 图 2(d)是 $P(c|X)$和 $P(X)$同时发生变化的情况,对应的是实质上的概念漂移[50].



(a) 原始数据 (a) Origin data　(b) 实质漂移($P(c|X)$变化) (b) $P(c|X)$ Changes　(c) 虚假漂移 (c) Virtual drift　(d) 实质漂移($P(c|X)$和$P(X)$变化) (d) $P(c|X)$ and $P(X)$ change

图 2　概念漂移类型

Fig. 2　The type of concept drift

2.1.2　常见概念漂移处理技术

漂移检测方法(DDM)[51]是概念漂移检测中最著名的代表方法. 它估计分类器误差(及其标准偏差),其(假设分类器训练方法的收敛)必须随着接收到更多训练样本而减少[52]. 如果分类器错误随着训练样本的数量而增加,则表明产生概念漂移,并且应该重新构建当前模型. 从技术上讲,如果估计误差加上其偏差的两倍达到警告水平,DDM 会产生警告信号. 如果达到警告级别,则会在特殊窗口中记住新的传入示例. 如果之后错误低于警告阈值,则此警告将被视为错误警报,并且此特殊窗口将被删除. 然而,误差随时间增加并达到漂移水平,当前分类器被丢弃,并且从存储在窗口中的最近标记的示例中学习新的分类器. 早期漂移检测方法(EDDM)是对 DDM 的修改,以优化渐变漂移的检测[7]. 通过比较错误率差别的新想法,实现了警告和漂移水平的相同概念. 另一个探测器 ECDD 采用观察指数加权移动平均值变化的机制[53]. PH 方法是一种基于序列分析检测器,可以有效地检测模型建立正常行为中的概念变化[46]. 累积和方法(CUSUM)是一种序列分析技术,通常被用来检测变化,而且不需要存储数据,并指出何时发生了显著

的变化[47]. 作为参数,可以考虑分类错误的期望值,其可以基于来自数据流传入示例的标签来估计. JOAO等人在文献[54]中提出对 CUSUM 参数与其性能之间的关系进行综合分析. SPC 是标准的统计技术,它是一种基于统计过程控制的检测器,把学习看做过程并且监督整个过程的改变[48]. ADWIN 使用 Hoeffding 边界来保证窗口的最大宽度,并且保证在窗口内没有概念变化. ADWIN 是比较两个滑动窗口方法中最著名的代表,是一种捕获流均值的很好方法. 在该算法[49]中,传入示例的窗口持续增长直到达到窗口的变化值. 当算法成功找到两个不同的子窗口时,它们的分裂点被认为是概念漂移的指示[36].

2.1.3　解决概念漂移的决策树算法

用于处理概念漂移的决策树算法包括 CVFDT、iOVFDT 等. VFDT 在充分的统计量和 Hoeffding 不等式(HB)的条件下,实现了启发式评价函数的节点分裂. 由于 VFDT 模型不能处理概念漂移,Hulten 等人提出了一个 VFDT 的优化版本,称为概念漂移快速决策树(CVFDT)[35]. 该算法能够适应数据流中的概念漂移,并且具有较高的分类精度和快速决策树的特点,在数据样本生成过程中具有检测和响应概念变化的能力. Liang 等人提出了一种基于 CVFDT 的不确定处理概念漂移快速决策树(UCVFDT)[28]算法,它是一种同时处理不确定性数据流和概念漂移的算法. 尽管当时已经提出了大量基于集成分类模型的归纳学习算法来处理概念漂移数据流,但是很少关注概念漂移的多样性检测以及数据流中噪声产生的影响. 受此启发,Li 等人提出了概念漂移随机决策树(CDRDT)算法[29],该算法利用随机决策树的集成模型来区分各种类型的概念漂移与噪声数据流. 使用可变的小数据块来逐步生成随机决策树. 同时引入了 Hoeffding 不等式和统计控制原理来检测不同类型的概念漂移和噪声数据流. 在虚拟和真实数据流的广泛研究中证明了 CDRDT 可以有效且高效地检测来自嘈杂数据流中的概念漂移. 针对网络流量存在概念漂移、不同应用类型数据流偏态分布等特性,张剑等人提出了基于 Hoeffding 不等式的自适应分级滑动窗口决策树(AGSW-DT)算法[30]. 该算法按照节点信息增益率检测概念漂移、动态调整概念漂移检测窗口及不同类型训练样本集的窗口,实现对不同速率概念漂移的自适应分类和决策树更新.

近 3 年出现的用于处理概念漂移问题的数据流决策树算法包括 AFDT、EDTC 和 CSPT 等. 由于不确定数据流中一般隐含着概念漂移问题,因此对其进行有效分类存在着很大困难. 为此,刘志军等人提出一种自适应快速决策树(AFDT)[31]算法,该算法能够快速地将不确定数据流的特征属性分类为不确定数值属性和不确定分类属性,并将两种属性与概念漂移问题紧密地联系起来,从而实现对不确定数据流的有效分类以及对其中隐含的概念漂移问题进行有效检测和处理. Farid 等人提出了一种算法,可以通过决策树来检测数据流中的新类[55]. 该方法将分类与聚类技术相结合,视概念漂移为一种新的类. 很少有在线分类算法基于传统的进化集成,如在线装袋或提升,专注于处理概念漂移的数据流,同时在嘈杂的数据流上表现良好. 基于传统归纳集成的在线分类算法很少关注概念漂移数据流的处理,而对噪声数据的处理效果较好. 在此基础上,Li 等人提出了一种基于概念漂移数据流的集成决策树(EDTC)增量算法[27]. 利用 Hoeffding 不等式中指定的两个阈值来区分概念漂移和噪声数据流,最终得出的结论是,提供多种解决方案来从噪声下的概念漂移数据流中学习. Krawczyk 等人提出一种自适应成本敏感的解决方案(CSPT)来处理不平衡和漂移的数据流[24],该解决方案能够从二进制和多类不平衡数据流中学习,而且还显示了如何使用滑动窗口以在线方式分析少数分类的结构. 这能够降低传入少数分类实例的难度,从而进一步了解当前数据流的状态.

2.2　增量模型处理技术

由于数据流的特征,对其进行处理时采用的主要方法是增量算法[54]. 增量学习(Incremental Learning)是指一个学习体系不断从新的样本数据中学习新知识的过程. 在数据流分类任务中,需要确保分类器能一直适用于当前的数据流分布,因此便需要获取新的数据对原始分类模型进行修改,这种往复不断学习新实例的方法是处理数据流问题不可或缺的[56]. 增量模型处理方法包括概念漂移快速决策树(CVFDT)、霍夫丁选项树(HOT)、霍夫丁自适应树(HAT)、增量优化快速决策树(iOVFDT)[22,25]、增量进化决策树(incremental evolutionary algorithm for decision tree induction, INEVOT)算法[16]、灵活模糊决策树(flexible decision tree, FlexDT)[57]、多灵活模糊决策树(multi-flexible fuzzy decision tree, MFlexDT)[58]和基于混合划分标准的多变量分支模糊决策树(multivariate branch fuzzy decision tree based on hybrid partitioning standard, MHFlexDT)算法[59]等.

上述所有方法都能适应概念漂移的数据流,但是在预测精度方面,特别是在处理噪声数据流时,它们不能很好地表现出来. 这是因为这些分类方法需要通知分裂测试. 这将导致更大的噪音数据的影响. Li 等人在文献[27]中提出了一种基于随机决策树的增量集成算法(EDTC),用于带噪声的概念漂移数据流. 这项工作是在以前工作的基础上发展起来的[60]. Jankowski 等人提出的增量决策树解决方案是 INEVOT[15],该算法是 EVO-Tree 算法的修改版本. INEVOT 类似于 EVO-Tree,它基于一种进化计算,但它使用滑动窗口来处理数据流. 该算法对树的种群进行处理. 当新的数据流到达时,通过进化算子对种群进行变换,并从中提取出最好的个体(树)进行分类. 与 EVO-tree 相比,INEVOT 算法的主要创新之处在于它的初始化过程,该过程允许存储先前收集的知识. 灵活模糊决策树(FlexDT)[57]是用于处理数据流的模糊决策树的最新版本. 它具有灵活性和增量性等优点,因此便于处理噪声和丢失的数值属性. 它使用二进制分类,所以它的灵活性还不够高,从而使得树的层数变大. 另一方面,二进制分类使 FlexDT 限制在每层管理多个范围,并具有更多的可选值. 后来提出了一种称为多灵活模糊决策树(MFlexDT)的方法. 将 FlexDT 扩展到多分区,从而使过程更加灵活,以处理更多可选值的范围[58]. Song 等人提出了基于混合划分标准的多变量分支模糊决策树(MHFlexDT)算法,该算法可以在降低模糊决策树层数的同时获得更高的分类精度,使属性操作值的范围灵活性更高,提高了分类的响应能力[59]. MHFlexDT 可以在决策树的不同层次上添加新特征,这可以使模糊决策树的构造更加灵活. 通过使用临时分支为具有低成员资格的数据提供新方法,并且该方法与二进制 HFlexDT 算法相比减少了数据计算成本. 另一方面,它更容易找到概念变化的数据,并且由于使用临时分支而创建新分支. 总之,基于增量学习的 MHFlexDT 算法有助于在具有非结构化数据的复杂环境中进行学习.

3　算法的分析与总结

为了更好地分析总结数据流决策树分类算法的性能,表 1 从分类模型、是否增量、是否可以处理概念漂移、实验数据流和算法优缺点等几个角度分析总结决策树算法. 总体来说,VFDT 是处理数据流分类中最经典的决策树算法,它开启了数据流决策树分类算法的先河,该算法也是第一个基于 Hoeffding 不等式设计的决策树算法,但是该模型不能处理概念漂移. 相继 VFDT 之后的大多数数据流决策树算法都是基于 Hoeffding 不等式设计的,同时这些决策树算法都能处理概念漂移,并且分类性能都很好. 目前分类性能较好,并且应用广泛的决策树算法有以下几种:MHFlexDT、RDT、iOVFDT、UCVFDT、EDTC 和 EFDT 等. 算法性能比较结果如表 1 所示.

表 1　决策树算法性能比较

Table 1　Performance Comparison of Decision Tree Algorithms

算法	分类模型	增量	概念漂移	实验数据	优缺点
HoeffdingTree	单分类	是	否	Web data	优点:最先使用 Hoeffding 不等式处理数据流 缺点:不适用于噪声数据流;有固定的主动分裂阈值
CVFDT	单分类	是	是	Rotating Hyperplane、Web data	优点:使用滑动窗口来处理概念漂移 缺点:突变概念漂移的情况下;很难快速检测到概念漂移
VFDTc	单分类	是	是	Waveform21、Waveform40、LED	优点:提供功能树叶 缺点:由于树大小暴增;不适用于实际应用
UFFT	集成分类	是	是	Balance dataset、Waveform21、Waveform40、LED	优点:单次扫描算法,树森林检测概念漂移 缺点:为每个可能的类构建二叉树,而不是单一的树归纳方法
HOT	集成分类	是	是	GENF1-F10、RTS、RTC、LED、Covertype、Waveform21、Waveform40、RRBFS、RRBFC	优点:提供可选的子树;高精度的后剪枝 缺点:树大小暴增并且计算速度慢
OcVFDT	单分类	是	是	—	优点:将一分类器与 VFDT 相结合 缺点:不能处理概念漂移
FlexDT	集成分类	是	是	—	优点:使用 sigmod 函数来处理不完美的流;FlexDT 抗噪声能力强;FlexDT 能够有效地适应新的概念;FlexDT 可以对缺失值的实例进行分类 缺点:计算速度慢;当数据中包含名称属性时,FlexDT 的灵活性降低;使用多个分区会导致更高的时间复杂度和更高的分类方差

续表 1　Table 1 continued

算法	分类模型	增量	概念漂移	实验数据	优缺点
MFlexDT	集成分类	是	是	LED、RTS、RTC、Electricity、Airlines	优点:对数值特征(变量)和标称特征(变量)都能生成最优树;具有多个分区,可以进行自动在线模糊数据分类
MHFlexDT	集成分类	是	是	Waveform、Airlines	优点:可以减少系统计算,降低模糊决策树深度的同时获得更高的精度. 该算法使属性操作值的范围更加灵活,提高了分类响应性
FPDT	集成分类	是	是	SEA Concepts、Hyperplane、Random RBF、LED、Covertype、Poker-Hand、Electricity	优点:运行时间减少,且保持高准确率 缺点:使用灵活的学习速率可以获得更准确的方法,可能会产生大量额外的运行时或内存开销
RDT	集成分类	是	是	KDDCUP'98	优点:训练效率高,内存需求明显小于学习单个最优树
EFMMDT	单分类	是	是	Hyperplane、RBF、SEA、Waveform、Elec、ForestCovtype、Letter、SatelliteImage、Shuttle	优点:不仅提供了流上下文所需的灵活性,而且提高了稳定性;使得生成的决策树更小、深度更浅;特别是存在概念漂移时,算法具有更高的精度
RF	集成分类	是	是	Breast cancer、Diabetes、Sonar、Liver 等	优点:对于较大的数据集流,可以显著降低错误率
SRF	集成分类	是	是	Synthetic data、Forest CoverType	优点:该算法的分类精度与标准随机森林算法相当;每条记录分类时间复杂度为 $O(t)$,其中 t 为森林中的树数
DSRF	集成分类	是	是	—	优点:使用基于熵的漂移检测技术处理不断发展的数据流;该算法可以处理基类边界漂移的多类问题,而不会失去准确性
SRMTDS	集成分类	是	是	Kddcup99、WaveForm40、LED	优点:VFDTc 相比,SRMTDS 在时间、空间、精度和抗噪声能力方面具有更高的性能 缺点:仍然需要创建多个决策树,因此它在独立环境中具有一定的劣势;如何进一步减少运行时间,改进多树分类阶段的交互投票机制,以及如何处理随时间推移的概念漂移是下一步的工作
MSRT	集成分类	是	是	HyperPlane、STAGGER、KDD-Cup99	优点:与 CVFDT 相比,MSRT 在时间、精度和鲁棒性方面都有了提高
GDT	单分类	是	是	Synthetic data	优点:GDT 算法在时间消耗方面明显优于 ID3;在分类精度方面显著优于 McDiarmid 树算法
CDRDT	集成分类	是	是	HyperPlane、SEA、STAGGER、KDDCup99、Yahoo! shopping	优点:CDRDT 能够有效地检测出噪声流数据中的概念漂移;对噪声具有较强的鲁棒性;CDRDT 在运行时开销和空间开销方面都有明显的优势,且预测精度没有任何损失 缺点:不能辨别模型噪声数据的概念漂移噪声;不能适应周期性概念漂移
iOVFDT	集成分类	是	是	LED、Wave、RTS、RTC	优点:iOVFDT 具有更高的精度和更紧凑的树大小,动态减少学习时间和错误率
EDTC	集成分类	是	是	Covertype、LED、Waveform	优点:该方法能有效地解决概念漂移数据流在噪声环境下的学习问题 缺点:Hoeffding 不等式只能处理数值数据;分裂度量方面存在不足;模型应用到带有标记数据和未标记数据的数据流中是否适用
DTFA	单分类	是	是	RandomTree	优点:算法在数值数据流实验中取得了较好的分类精度
CEVOT	单分类	是	是	Airlines、CovtypeNorm、elec、kddcup99、LED、Wave 等	优点:该算法在准确性和处理时间方面具有较高性能
EVO-Tree	单分类	是	是	Abalone、ecoli、page-blocks、winequality-red、breast tissue、seeds	优点:该算法在减少树大小的同时减小了分类误差
UCVFDT	集成分类	是	是	Hyperplane、SEA、Forest Covertype	优点:既保持了 CVFDT 对概念漂移的高速处理能力,又增加了对不确定性数据的处理能力;UCVFDT 算法能够有效地对具有不确定数值属性的动态数据流进行分类,具有较高的计算效率　缺点:数据流高度不确定性
CSPT	集成分类	是	是	Artificial、Twitter、LED、Hyperplane、RBF、RTree、poker	优点:可用于处理在线不平衡类,处理概念漂移有效且快速　缺点:其预测能力和漂移处理能力有待提高
AFDT	单分类	是	是	Covertype	优点:具有很强的对无标记不确定数据流概念漂移的处理能力;该算法适用于大量无标记不确定数据流概念漂移的检测和分类
EFDT	单分类	是	是	KDD、WISDM、Poker、Fonts、Forest Covertype、Skin 等	优点:EFDT 能够很快地学习所有的概念,并在观察到新的例子时不断调整潜在的过拟合现象　缺点:处理数据运行时间相比 VFDT 算法较长

4 下一步工作

概念漂移问题在近十几年来得到了一定的发展,但是现有的方法仍然存在很多不足之处,这为学者下一步的研究提供了方向. 多个数据流的表示也是现在的一个热点问题,而且目前还没有针对在线问题提出纯粹的基于包装器的解决方案,这将是数据流面临的一大挑战.

(1)对于能够直接解决概念漂移问题的特征和实例选择方法需要进一步研究. 解决这个问题的一种方法是将实例选择方法与漂移检测模块结合起来,这将直接影响原有训练模型的可用性. 另一种可能的解决方案是使用加权模型. 这样就可以顺利地遗忘过时的模型,同时保留其中有用的部分. 仅在分类子集内发生的局部漂移也应该被考虑. 这种情况下,只有在漂移存在的区域中才需要修改选定的模型. 这将需要基于类模型的修剪方法和忽略这些漂移模型对静态类影响的方法[61].

(2)处理多个流和更复杂的表示:几乎所有的数据流集成处理器都被提议只处理单个数据流. 然而,一些应用程序(例如,在生存分析变体[62]中)对互联网信息或审查数据的研究可以提供几个并行的流. 在如此多的数据流中,相同的数据事件可能出现在每个流中的不同时刻,并且可能具有不同的描述. 这带来了一些有趣的新挑战. 在将大数据分析[63]中的不同(异构)数据存储库进行集成时,这些方面应该特别重要. 在一些新应用程序中,数据流变得越来越复杂,它们需要在同一时刻处理许多异构的数据表示. 这种混合的表示包括结构化的、半结构化的和完全非结构化的数据字段. 为了充分理解这些数据源的动态和现象,研究者需要找到这些复杂数据变化之间的交互. 综上所述,综合各种不同的处理模型,它们似乎是解决这一挑战的极具前景的解决方案.

(3)目前还没有针对在线问题提出纯粹的基于包装器的解决方案. 这些方法的有效实施可能会对其增加的计算成本造成挑战,但这可能会被在线学习者固有的辨别能力和它们的适应能力所弥补. 一个潜在的解决方案是将过滤器和包装器方法结合起来,以减少使用更昂贵方法的次数,并且即使在包装器计算过程中也允许连续地分类. 另一个潜在的解决方案是使用基于 GPU 或分布式计算的高性能解决方案,以减少与此方法相关的计算负载[64].

[参考文献]

[1] GHAZIKHANI A, MONSEFI R, YAZDI H S. Ensemble of online neural networks for non-stationary and imbalanced data streams[J]. Neurocomputing 2013, 122: 535-544.

[2] CAO K Y, WANG G R, HAN D H, et al. An algorithm for classification over uncertain data based on extreme learning machine[J]. Neurocomputing, 2016, 174(Part A): 194-202.

[3] CERVANTES J, LAMONT F G, CHAU A L, et al. Data selection based on decision tree for SVM classification on large datasets[J]. Applied soft computing, 2015, 37: 787-798.

[4] KRANJC J, SMAILOVIC J, PODECAN V. learning for sentiment analysis on data streams: methodology and workflow implementation in the ClowdFlows platform[J]. Information processing & management, 2015, 51(2): 187-203.

[5] RUTKOWSKI L, JAWORSKI M, DUDA P. Decision Trees in Data Stream Mining[M]//Stream Data Mining: Algorithms and Their Probabilistic Properties. Switzerland: Studies in Big Data, Springer Nature Switzerland AG 2020: 37-50.

[6] COSTA V G T D, CARVALHO A C, JUNIOR S B. Strict very fast decision tree: a memory conservative algorithm for data stream mining[J]. Pattern Recognition Letters, 2018, 116: 22-28.

[7] 丁剑,韩萌,李娟. 概念漂移数据流挖掘算法综述[J]. 计算机科学,2016,43(12):24-29.

[8] MOHAMED M G, ARKADY Z, SHONALI K. A survey of classication methods in data streams[J]. Springer U, 2007, 43(2): 39-59.

[9] BRZEZINSKI D, STEFANOWSKI J. Combining block-based and online methods in learning ensembles from concept drifting data streams[J]. Information sciences, 2014, 265(5): 50-67.

[10] ABBASZADEH O, AMIRI A, KHANTEYMOORL A R. An ensemble method for data stream classification in the presence of concept drift[J]. Front inform technol electron Eng, 2015, 16(12): 1059-1068.

[11] RUTKOWSKI L, JAWORSKI M, PIETRUCZUK L, et al. A new method for data stream mining based on the misclassification error[J]. IEEE transactions on neural networks & learning systems, 2015, 26(5): 1048-1059.

[12] RUTKOWSKI L, JAWORSKI M, PIETRUCZUK L, et al. Decision trees for mining data streams based on the Gaussian approx-

imation[J]. IEEE transactions on knowledge & data engineering,2013,26(1):108-119.
[13] RUTKOWSKI L,JAWORSKI M,PIETRUCZUK L,et al. The CART decision tree for mining data streams[J]. Information sciences,2014,266(5):1-15.
[14] RUTKOWSKI L,PIETRUCZUK L,DUDA P,et al. Decision trees for mining data streams based on the McDiarmid[J]. IEEE transactions on knowledge & data engineering,2013,25(6):1272-1279.
[15] JANKOWSKI D,JACKOWSKI K. Evolutionary algorithm for decision tree induction[C]//IFIP International Conference on Computer Information Systems and Industrial Management. Berlin,Heidelberg:Springer,2014:23-32.
[16] JANKOWSKI D,JACKOWSKI K. An increment decision tree algorithm for streamed data[C]//Trustcom/bigdatase/ispa. Helsinki,Finland:IEEE,2015:199-204.
[17] JANKOWSKI D,JACKOWSKI K,CYGANEK B. Learning decision trees from data streams with concept drift[J]. Procedia computer science,2016,80:1682-1691.
[18] MIRZAMOMEN Z,KANGAVARI M R. Evolving fuzzy min-max neural network based decision trees for data stream classification[J]. Neural processing letters,2017,45(1):341-363.
[19] DUDA P,JAWORSKI M,PIETRUCZUK L,et al. A novel application of Hoeffding's inequality to decision trees construction for data streams[C]//International Joint Conference on Neural Networks. Beijing:IEEE,2014:3324-3330.
[20] 陈煜,李玲娟. 一种基于决策树的隐私保护数据流分类算法[J]. 计算机技术与发展,2017,27(7):111-114.
[21] MANAPRAGADA C,WEBB G,SALEHI M. Extremely fast decision tree[C]//International Conference on Knoledge Discovery,London,Unite Kingdom,2018:1-10.
[22] CZARNOWSKI I,JEDRZEJOWICZ P. Ensemble classifier for mining data streams[J]. Procedia computer science,2014,35(9):397-406.
[23] HAN D,LI S,WEI F,et al. Two birds with one stone:classifying positive and unlabeled examples on uncertain data streams[J]. Neurocomputing,2018,277(1):149-160.
[24] KRAWCZYK B,SKRYJOMSKI P. Cost-sensitive perception decision trees for imbalanced drifting data streams[M]//Machine Learning and Knowledge Discovery in Databases. Cham:Springer,2017:512-527.
[25] YANG H,FONG S. Incrementally optimized decision tree for mining imperfect data streams[M]//Networked Digital Technologies. Berlin,Heidelberg:Springer,2012:281-296.
[26] ZLIOBAITE I. Learning under concept drift:an overview[J]. Computer science,2010,270(10):1-36.
[27] LI P P,WU X D,HU X G. Learning concept-drifting data streams with random ensemble decision trees[J]. Neurocomputing,2015,166(C):68-83.
[28] LIANG C Q,ZHANG Y,SONG Q. Decision tree for dynamic and uncertain data streams[C]//Proceedings of the Second Asian Conference on Machine Learning. ACML,Tokyo,Japan:Microtome Publishing,2010:209-224.
[29] LI P P,WU X D,HU X G,et al. A random decision tree ensemble for mining concept drifts from noisy data streams[J]. Applied artificial intelligence,2010,24(7):680-710.
[30] 张剑,曹萍,寿国础. 网络流量识别的自适应分级滑动窗决策树算法[J]. 计算机应用研究,2013,30(8):2470-2472.
[31] 刘志军,张杰,许广义. 基于自适应快速决策树的不确定数据流概念漂移分类算法[J]. 控制与决策,2016,31(9):1609-1614.
[32] DOMINGOS P,HULTEN G. Mining high-speed data streams[C]//Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Boston:ACM Press,2000:71-80.
[33] HULTEN G,DOMINGOS P. Mining decision trees from streams[M]//Data Stream Management. Berlin Heidelberg:Springer,2016.
[34] JOAO G,ROCHA R,MEDAS P. Accurate decision trees for mining high-speed data streams[C]//Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Washington,DC,USA:ACM,2003:24-27.
[35] HULTEN G,SPENCER L,DOMINGOS P. Mining time-changing data streams[C]//Proceeding of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Fransisco,2001:97-106.
[36] KRAWCZYK B,MINKU L L,GAMA J. Ensemble learning for data stream analysis:a survey[J]. Information fusion,2017,37(C):132-156.
[37] PFAHRINGER B,HOLMES G,KIRKBY R. New options for Hoeffding trees[M]//AI 2007:Advances in Artificial Intelligence. Berlin Heidelberg:Springer,2007.
[38] BIFET A,HOLMES G,PFAHRINGER B,et al. Fast perceptron decision tree learning from evolving data streams[C]//Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining. Berlin:Springer,2010:299-310.
[39] FAN W,WANG H,YU P S,et al. Is random model better? On its accuracy and efficiency[C]//IEEE International Conference on Data Mining. New York:Hawthorne,IEEE,2003:51-58.

[40] BREIMAN L. Random forests[J]. Machine learning,2001,45(1):5-32.
[41] GAMA J,MEDAS P,POCHA R. Forest trees for on-line data[C]//Proceedings of ACM Symposium on Applied Computing. ACM,New York,NY,USA,2004:632-636.
[42] ABDULSALAM H,SKILLICORN D B,MARTIN P. Streaming random forests[C]//Database Engineering and Applications Symposium,2007. Ideas 2007 International. Banff,Alta,Canada,IEEE,2007:225-232.
[43] HU X,LI P,WU X,WU G. A semi-random multiple decision-tree algorithm for mining data streams[J]. Journal of Computer Science and Technology,2007,22(5):711-724.
[44] ABDULSALAM H,SKILLICORN D B,MARTIN P. Classifying Evolving Data Streams Using Dynamic Streaming Random Forests[C]//International Conference on Database and Expert Systems Applications. Kingston,Canada:Springer-Verlag,2008:643-651.
[45] LI P,HU X,WU X. Mining concept-drifting data streams with Multiple Semi-Random Decision Trees[C]//International Conference on Advanced Data Mining and Applications. Berlin,Heidelberg:Springer-Verlag,2008:733-740.
[46] IKONOMOVSKA E,GAMA J,DEROSKI S. Learning model trees from evolving data streams[J]. Data mining & knowledge discovery,2011,23(1):128-168.
[47] JABER G,CORNUEJOLS A,TARROUX P. A new on-line learning method for coping with recurring concepts:The ADACC System[C]//International Conference on Neural Information Processing. Berlin,Heidelberg:Springer,2013:595-604.
[48] ZHUKOV A V,SIDOROV D N,FOLEY A M. Random forest based approach for concept drift handling[C]//Analysis of Images Social Networks and Texts:5th International Conference. Yekaterinburg,Russia,2016,661:69-77.
[49] BIFET A,GAVALDA R. Learning from time-changing data with adaptive windowing[C]//Siam International Conference on Data Mining. Minneapolis:DBLP,2007.
[50] 白洋. 数据流概念漂移检测和不平衡数据流分类算法研究[D]. 北京:北京交通大学,2017.
[51] GAMA J,MEDAS P,CASTILLO G,et al. Learning with drift detection[J]. Intelligent data analysis,2004,8:286-295.
[52] RAUDYS S. Statistical and neural classifiers:an integrated approach to design[M]. Berlin,Heidelberg:Springer-Verlag,2014:289.
[53] ROSS G J,ADAMS N M,TASOULIS D K,et al. Exponentially weighted moving average charts for detecting concept drift[J]. Pattern recognition letters,2012,33(2):191-198.
[54] JOAO G,BIFET A,PECHENIZKIY M,et al. A survey on concept drift adaptation[J]. Acm computing surveys,2014,46(4):1-37.
[55] FARID D M,RAHMAN C M. Novel class detection in concept-drifting data stream mining employing decision tree[C]//International Conference on Electrical & Computer Engineering. Dhaka,Bangladesh:IEEE,2013:630-633.
[56] BARDDAL J P,GOMES H M,ENEMBRECK F. A survey on feature drift adaptation[C]//IEEE International Conference on Tools with Artificial Intelligence. Vietri Sul Mare,Italy:IEEE,2015:1-8.
[57] HASHEMI S,YANG Y. Flexible decision tree for data stream classification in the presence of concept change,noise and missing values[J]. Data mining & knowledge discovery,2009,19(1):95-131.
[58] ISAZADEH A,MAHAN F,PEDRYCZ W. MFlexDT:multi flexible fuzzy decision tree for data stream classification[J]. Soft computing,2016,20(9):3719-3733.
[59] SONG X,WANG H,HE H Y,et al. MHFlexDT:a multivariate branch fuzzy decision tree data stream mining strategy based on hybrid partitioning standard[C]//International Symposium on Neural Networks. Cham:Springer,2018:310-317.
[60] LI P P,WU X D,LIANG Q H,et al. Random ensemble decision trees for learning concept-drifting data streams[M]//Advances in Knowledge Discovery and Data Mining. Berlin,Heidelberg:Springer,2011:313-325.
[61] RAMREZ G S,KRAWCZYK B,CARCIA S,et al. A survey on data preprocessing for data stream mining[J]. Neurocomputing,2017,239(C):39-57.
[62] JAPKOWICZ N, STEFANOWSKI J. Big data analysis: new algorithms for a new society[M]. Switzerland: Springer International Publishing,2016:1-10.
[63] SHAKER A,HULLERMEIER E. Survival analysis on data streams:analyzing temporal events in dynamically changing environments[J]. International journal of applied mathematics & computer science,2014,24(1):199-212.
[64] CANO A, ZAFRA A. Solving classification problems using genetic programming algorithms on GPUs[C]//International Conference on Hybrid Artificial Intelligence Systems. Berlin,Heidelberg:Springer-Verlag,2010:17-26.

[责任编辑:顾晓天]