ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 37,No. 5,Sep. 2022,pp. 997－1011
DOI:10. 16337/j. 1004-9037. 2022. 05. 005

http://sjcj. nuaa. edu. cn
E-mail:sjcj@nuaa. edu. cn
Tel/Fax:＋86-025-84892742

基于串行交叉混合集成的概念漂移检测及收敛方法

郭虎升[1,2]，高淑花[1]，王文剑[1,2]

(1. 山西大学计算机与信息技术学院,太原 030006; 2. 计算智能与中文信息处理教育部重点实验室(山西大学),太原 030006)

摘　要:概念漂移处理大多采用集成学习策略,然而这些方法多数不能及时提取漂移发生后新分布数据的关键信息,导致模型性能较差。针对这个问题,本文提出一种基于串行交叉混合集成的概念漂移检测及收敛方法(Concept drift detection and convergence method based on hybrid ensemble of serial and cross,SC_ensemble)。在流数据处于平稳状态下,该方法通过构建串行基分类器进行集成,以提取代表数据整体分布的有效信息。概念漂移发生后,在漂移节点附近构建并行的交叉基分类器进行集成,提取代表最新分布数据的局部有效信息。通过串行基分类器和交叉基分类器的混合集成,该方法兼顾了流数据包含的整体分布信息,又强化了概念漂移发生时的重要局部信息,使集成模型中包含了较多“好而不同”的基学习器,实现了漂移发生后学习模型的高效融合。实验结果表明,该方法可使在线学习模型在漂移发生后快速收敛,提高了模型的泛化性能。

关键词:流数据;概念漂移;集成学习;串行分类器;交叉分类器;混合集成

中图分类号:TP18　　**文献标志码**:A

Concept Drift Detection and Convergence Based on Hybrid Ensemble of Serial and Cross

GUO Husheng[1,2], GAO Shuhua[1], WANG Wenjian[1,2]

(1. School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China; 2. Key Laboratory of Computational Intelligence and Chinese Information Processing(Shanxi University), Ministry of Education, Taiyuan 030006, China)

Abstract: Concept drift is an important and difficult issue in streaming data mining tasks. At present, the concept drift processing methods adopt the ensemble learning strategy mostly. However, most of these methods cannot extract the key information of the new data distribution after concept drift, leading to poor model performance. To solve this problem, this paper proposes a concept drift detection and convergence method based on hybrid ensemble of serial and cross (SC_ensemble). When streaming data are in a stable state, the method trains serial base classifiers for ensemble learning, to extract effective information representing the overall data distribution. After concept drift occurs, parallel cross base classifiers are constructed near the drift site for ensemble learning, to extract the local effective information representing the latest data distribution. By ensemble learning of serial base classifiers and cross classifiers, the method takes into account the overall distribution information contained in streaming data, and strengthens the

基金项目:国家自然科学基金(62276157,U21A20513,62076154,61503229);中央引导地方科技发展资金(YDZX20201400001224);山西省自然科学基金(201901D111033);山西省重点研发计划项目(国际合作)(201903D421050)。
收稿日期:2021-09-16;**修订日期**:2022-01-01

important local information when concept drift occurs, so that the ensemble model contains more "good but different" base learners, and realizes the efficient combination of learning models after concept drift. The experimental results show that the proposed method can make the online learning model converge quickly after concept drift, and improve the generalization performance of the model.

Key words: streaming data; concept drift; ensemble learning; serial classifier; cross classifier; hybrid ensemble

引　　言

大数据时代,动态数据不断在各大应用领域涌现,如交通数据、网页点击和股票预测等。与传统静态数据相比,这些数据具有高速、实时、多变以及不可预知等特点,故将这样的数据称之为流数据[1-2]。近年来,流数据挖掘受到了越来越多的关注,其目的是使学习模型能更准确地预测数据分布变化,提高在线学习模型的泛化性能[3-5]。在流数据中,数据分布的不稳定性和动态变化等特征会导致数据分布以及其中隐含的目标概念随着环境等因素的改变而发生变化,即概念漂移[6-7]。概念漂移是指在给定的输入特征下,输出的目标概念发生了改变。其中,概念可以理解为某一时刻所有样本的空间分布,要学习的概念或者函数被称为目标概念,可以用数据的联合概念分布$P(\boldsymbol{x},y)$表示,其中$\boldsymbol{x}$表示d维特征向量,y表示相对应样本的标签。若当前时刻t发生了概念漂移,可以将其形式化表示为:$\exists \boldsymbol{x}$: $P_{t-1}(\boldsymbol{x},y)\neq P_t(\boldsymbol{x},y)$。然而,在流数据挖掘中,由于概念漂移的存在,使得传统机器学习方法无法满足高实时泛化性能的需求。如在垃圾邮件过滤中,客户喜好变化会改变垃圾邮件的定义,垃圾邮件的范畴也可能会随时间而变化;在天气预报中,天气情况可能会随温度、压强和湿度等因素而改变。此时,需要搜集大量的实时数据并不断调整模型,才可能提高模型的泛化性能[8]。因此,在流数据挖掘中,提高概念漂移发生后在线学习模型的收敛性能具有重要意义。

概念漂移发生后,集成学习策略能够灵活更新基分类器,有效提高模型的泛化性能。因此,集成学习是解决概念漂移的有效途径,然而由于漂移刚发生时得到的新分布数据有限,集成模型中会保留较多携带旧分布数据信息的基分类器,大多数基分类器性能较差,不能达到"好而不同"的集成,导致集成模型泛化性能较差。为提高概念漂移流数据挖掘的性能,本文提出了一种基于串行交叉混合集成的概念漂移检测及收敛方法。在流数据处于平稳状态下,该方法通过构建串行基分类器进行集成,以提取代表数据整体分布的有效信息。概念漂移发生后,在漂移节点附近构建并行的交叉基分类器进行集成,提取代表最新分布数据的局部有效信息。

本文主要贡献包括:(1)通过串行基分类器和交叉基分类器的混合集成,该方法兼顾了流数据包含的整体分布信息,又强化了概念漂移发生时的重要局部信息。(2)在集成模型中增加了较多"好而不同"的基学习器,实现了漂移发生后在线集成学习模型的高效收敛,提升了模型泛化性能。在10个数据集上与5种对比方法进行比较,实验结果显示大部分数据集上本文方法的平均实时精度和累计精度均高于其他方法,能在概念漂移发生后具有较高的恢复性能,且相比于其他方法,本文方法的整体鲁棒性值最大,有效提高了模型的泛化性能。

1　相关工作

目前,对于流数据挖掘中常存在的概念漂移问题,常见的处理策略主要包括基于实例选择的方法和基于集成学习的方法。基于实例选择的方法通常采用滑动窗口技术来实现,即通过使用一个或多个滑动窗口来存储数据,通过不断向前滑动窗口来判断是否发生概念漂移,选择最新的样本进行模型训练和更新,以保证当前窗口的数据能够服从最新分布。典型方法如:基于双窗口滑动的漂移检测方

法[9-10],基于决策树算法调节分支节点模型方法[11],基于信息熵判定的自适应滑动窗口方法[12],基于霍夫丁不等式的概念漂移检测方法[13]以及自适应滑动窗口概念漂移检测方法[14]。这些方法通过引入概念漂移检测机制,在一定程度上提高了在线学习模型的实时性能,但滑动窗口大小的调节是一个困难的问题,这在一定程度上影响了模型性能。

基于集成学习的方法通过构建一系列弱分类器,使用一定的相关组合规则将这些弱分类器进行组合,通过投票或者加权集成方式得出非稳态环境下预测性能较好的强分类器。基于集成学习的方法又可以分为基于数据块的集成与在线集成。

基于数据块的集成将流数据划分成固定大小的数据块进行处理,最常见的方式是构建有限数量的基分类器,根据一定规则用最新数据块上创建的分类器替换集成分类器中表现性能较差的分类器,典型方法包括:基于传统流数据的集成分类方法[15],该方法在连续的数据块上构建基分类器,并且使用启发式替换策略组合成固定大小的集成。基于动态调整基分类器权重的方法[16-19],此类方法通过不断动态调整各个基分类器的权重来适应概念漂移。基于选择性集成的在线自适应深度神经网络方法[20],该方法通过浅层次特征与深层次特征进行结合组成自适应深度单元,根据流数据的变化情况,动态调整网络中的信息流。虽然基于数据块的集成分类方法在很大程度上能提高模型的整体预测性能,但在概念漂移发生后,较多过时的基分类器会降低模型的效果。

在线集成是对样本进行逐一处理的集成方法。典型方法包括:基于动态加权投票的概念漂移适应方法[21-22],此类方法根据对新样本的预测准确性初始化权重,并且根据全局预测和局部预测来更新权重,以此来动态更新基分类器。基于单样本增量模型方法[23],该方法首先初始化一组基分类器,根据每个时间戳到达的单个样本更新集成模型,并进行分类器的加权组合。基于混合标记策略的在线学习方法[24],该方法的集成分类器由固定基分类器和动态基分类器组成进行概念漂移的适应,以及对Online bagging方法[25-26]进行改进的Leveraging bagging方法[27]。与基于数据块的集成方法相比,在线集成能够有效提高模型的实时性能,但由于其需要对样本逐一处理,因此学习效率较低。

本文提出了一种基于串行交叉混合集成的概念漂移检测及收敛方法,和传统方法相比,该方法在概念漂移发生后,有效提取了最新分布的关键信息;通过交叉分类器和串行分类器的集成方式进行模型融合,在保证整体实时性能的同时加快了模型的收敛速度。

2 基于串行交叉混合集成的方法

本文提出基于串行交叉混合集成的概念漂移检测及收敛方法,借助集成学习技术提高概念漂移发生后在线学习模型的性能。在平稳流数据环境下使用串行集成进行数据分布的整体信息提取。反之,在概念漂移发生后,使用交叉集成进行数据分布局部有效信息的提取,在保持数据整体分布有效信息的同时兼顾数据分布发生改变后的局部有效信息,提高了模型的泛化性和收敛性。SC_ensemble方法的整体框架图如图1所示。表1汇总了本文使用的符号及对应描述。

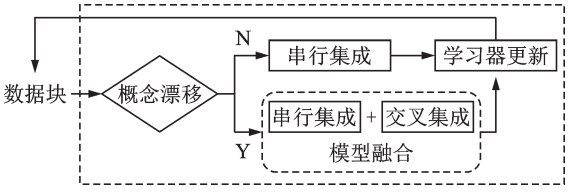

图1 SC_ensemble方法整体框架图

Fig.1 Overall framework of SC_ensemble algorithm

2.1 基分类器的串行集成

在流数据挖掘中,组合多个基分类器完成分类任务通常会比单个分类器得到更好的效果,这是由于串行生成的多个基分类器可以提取不同位点的数据信息,避免了流数据的局部微小波动带来的性能下降问题。对于串行的流数据集成学习过程,本文首先做以下约定:

表1 本文使用的符号汇总表

Table 1 Summary table of symbols used in this paper

符号	描述	符号	描述
t	当前时间戳	w	数据块的大小
D_t	当前时刻的数据块	D_i^*	第i个划分后的子数据块
H_t	当前时刻的串行分类器组合	H_t^*	当前时刻的交叉分类器组合
h_j	串行基分类器	h_j^*	交叉基分类器
k	串行基分类器的上限	f	交叉基分类器的上限
$\tilde{P}_t$	漂移位点检测波动比	$\tilde{Q}_t$	漂移结束检测波动比
w_{tj}	i时刻第j个分类器的权重	w_i	最新分类器的权重
C	惩罚因子	δ	概念漂移警告

(1) 假定流数据$\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_t,\cdots\}$,对数据单元设定阈值为$w$(即每个数据单元的样本容量),当样本数量达到阈值时,自动划分为下一个数据块,当前时刻的数据块记为$D_t=\{(\boldsymbol{x}_{tj},y_{tj})\}_{j=1}^{w_t}$,$y_{tj}$为样本$\boldsymbol{x}_{tj}$对应的真实数据标签。

(2) 设定当前时刻的基分类器组合为H_t(即在当前时刻发挥作用的基分类器组合),其中包含的基分类器上限为k。

当数据块D_t到达时,首先根据当前的集成分类器H_{t-1}对其进行预测,根据式(1~3)对集成中每一个基分类器进行权重更新[17],有

$$\mathrm{MSE}_{tj}=\frac{1}{|D_t|}\sum_{\{x,y\}\in D_t}(1-h_y^j(\boldsymbol{x}))^2 \tag{1}$$

$$\mathrm{MSE}_r=\sum_y p(y)(1-p(y))^2 \tag{2}$$

$$w_{tj}=\frac{1}{\mathrm{MSE}_r+\mathrm{MSE}_{tj}} \tag{3}$$

式中:MSE_{tj}表示h_t中每个基分类器在当前时刻t最新到达的数据块D_t上的预测错误率;h为正确分类的示性函数,然后对D_t训练得到基分类器h_t;MSE_r表示分类器的均方误差;p为各标签的分类概率;w_{tj}为当前时刻每个分类器的权重。

若最新训练得到的基分类器h_t具有较好的预测性能,应该给与较高的权重,根据式(4)对基分类器进行权重初始化,有

$$w_i=\frac{1}{\mathrm{MSE}_r} \tag{4}$$

若此时H_{t-1}中基分类器的个数没有超出分类器个数上限$k(j<k)$,则将h_t直接添加到H_{t-1}中,否则用新得到的基分类器h_t替换掉H_{t-1}中权重最小的基分类器,得到新的集成分类器H_t。

图2展示了串行集成的学习过程,图中立方体表示基分类器,带虚线的灰色立方体表示当前集成分类器H_{t-1}中权重最小的基分类器,当$k<K$时执行添加操作,否则执行替换操作。

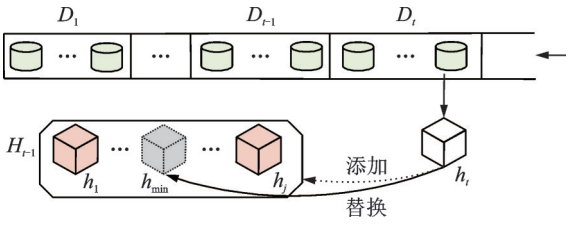

图2 串行集成学习过程

Fig.2 Serial ensemble learning process

2.2 基分类器的交叉集成

当流数据处于平稳状态时,不同时刻数据分布

差异不大,因此采用串行集成即可达到较好的分类效果。当概念漂移发生后,由于数据分布发生了较大变化,此时串行集成模型中存在大量分类性能很差的基分类器,在新数据块上训练表现较好的新分类器难以与串行集成中大量性能很差的过时基分类器相抗衡,因此很难对新概念做出快速响应,导致概念漂移发生后学习模型的泛化性能较差。因此,为了能够在概念漂移发生后提取到更多的新分布样本信息,本文在概念漂移发生后创建交叉分类器,在保留数据整体分布信息的同时提取概念漂移位点附近较多的局部分布信息,构建符合概念漂移环境下更好的集成学习模型,使在线集成学习模型在非稳定的流数据学习中得到更好的泛化性能。

2.2.1 概念漂移检测

为了能够在概念漂移发生后充分发挥交叉分类器的作用,则需要检测概念漂移发生的位点。当数据块D_t到来后,本文通过前两个位点得到的串行集成分类器H_{t-2}和H_{t-1}来检测是否发生概念漂移,H_{t-2}与H_{t-1}分别在数据块D_{t-1}和D_t上进行测试,得到测试精度Acc_{t-1}和Acc_t来表示,得出位点t的漂移位点检测波动比,有

$$\tilde{P}_t=\frac{\mathrm{Acc}_t}{\mathrm{Acc}_{t-1}} \tag{5}$$

随着流数据中新数据块的不断到达,当$\tilde{P}_t<\delta$(δ为概念漂移警告)时,表明集成模型在当前数据块上的测试精度发生显著下降,即数据分布发生了较大变化,因此触发概念漂移警告。图3为概念漂移检测过程。

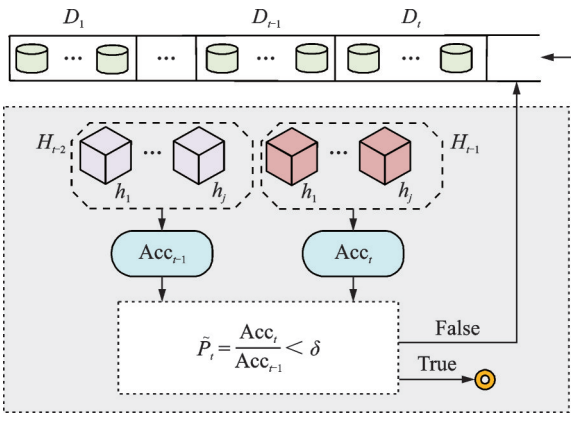

图3 概念漂移检测过程

Fig.3 Concept drift detection process

2.2.2 交叉分类器创建

当检测到可能发生概念漂移的位点后,在最新分布的数据上创建交叉分类器。该过程需要定义数据缓冲区和交叉集成,分别用D^*和H_t^*进行形式化表示。当检测到概念漂移位点后,将当前数据块D_t和前一个数据块D_{t-1}(D_{t-1}的加入包含两个作用:(1)方便交叉分类器构建;(2)避免概念漂移位点检测的时延,特别是对于渐变类概念漂移具有重要作用)加入数据缓冲区,即$D^*=D_{t-1}\cup D_t$,并在数据缓存区上构建交叉基分类器。

为了能够得到具有一定差异性的交叉基分类器,首先需要对数据缓存区D^*划分为不同的子数据块$D_i^*(i=1,\cdots,f)$,划分后的每个子数据块的大小也为w,且相邻的两个子数据块之间相隔一定数量的样本数s,其f与s之间服从式(6)的变化规律,然后对D_i^*进行训练得到基分类器h_i^*,分别将其添加至H_t^*中,得到交叉集成$H_t^*=\{h_1^*,\cdots,h_j^*,\cdots,h_f^*\}$,即

$$s=\left\lfloor\frac{w}{f}\right\rfloor+1 \tag{6}$$

当发生概念漂移后,串行集成中过时分类器占比较大,因此可能会在持续的一段时间内无法更好地适应新概念,此时需要不断构建交叉基分类器,以便模型更好地预测,直到数据变化较为平稳(即概念漂移结束)。为了找出漂移结束位点,本文将概念漂移发生时前一个时间戳的集成用H_A表示,在其对应数据块上的预测精度用Acc_A来表示,从此时刻起,H_A与Acc_A保持静止。随着流数据中新数据块的不断到达,需要进行漂移结束检测,其漂移结束检测波动比$\tilde{Q}_t$为

$$\tilde{Q}_t=\frac{\mathrm{Acc}_t}{\mathrm{Acc}_A} \tag{7}$$

若此时满足条件$\tilde{Q}_t<\delta$,则对数据缓冲区D^*重新进行子数据块的划分,从而更新交叉集成中的每

个基分类器。直到满足条件$\tilde{Q}_t \geqslant \delta$时，清空$D^*$，不再进行交叉基分类器的创建。此时刻起，恢复概念漂移的检测过程，根据漂移位点检测波动比寻找下一个概念漂移位点。

图4展示了交叉集成学习的过程。虚线框里的柱形表示对数据缓冲区划分的各个子数据块，分别由两部分组成，白色部分表示D_{t-1}中的数据，灰色部分表示D_t中的数据，带网格的剪角矩形和椭圆分别表示概念漂移位点的初始集成和该位点对应数据块的预测精度。

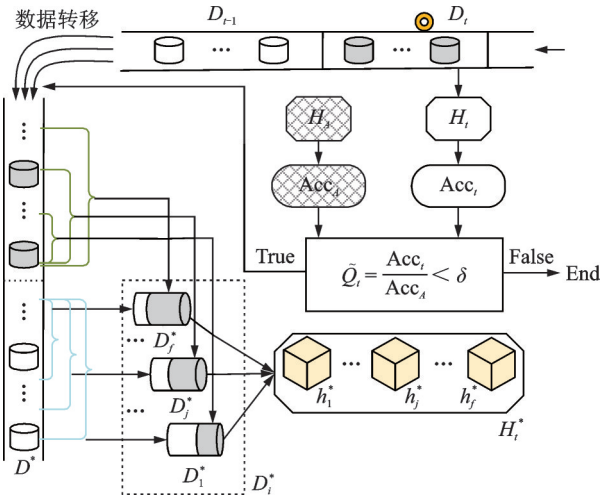

图4 交叉集成学习过程

Fig.4 Cross ensemble learning process

当检测到漂移发生后，由于漂移类型未知，漂移位点附近的样本都可能比较重要，因此本文通过串行基分类器和交叉基分类器的混合集成进行预测，即

$$H_{\mathrm{new}}(\boldsymbol{x})=c_{\arg\max}\left(\sum_{i=1}^{k}h_i^c(\boldsymbol{x})+\sum_{j=1}^{f}h_j^{*c}(\boldsymbol{x})\right) \tag{8}$$

可以看出，若交叉基分类器过多，模型容易出现过拟合，反之模型整体性能会被控制全局性能的串行集成所影响，未能达到预期的预测效果。

2.3 SC_ensemble算法

本文所提出的SC_ensemble算法借助集成学习的思想来检测和加速概念漂移的收敛，即通过检测是否有概念漂移的发生采用不同的处理方式，若检测到有概念漂移发生，则对缓冲区数据进行划分来创建交叉基分类器，很好地适应了流数据中的概念漂移。整体算法实现过程如图5所示。基于串行交叉混合集成的概念漂移检测及收敛算法如下所示。

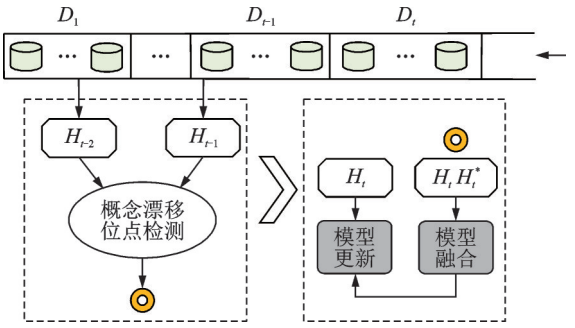

图5 SC_ensemble算法的整体实现过程

Fig.5 Overall implementation process of SC_ensemble algorithm

算法1 SC_ensemble算法

初始化：流数据序列$SD=\{D_i\}_{i=1}^{t}$，其中学习单元$D_i=\{(\boldsymbol{x}_{ij},y_{ij})\}_{j=1}^{w}$；数据块大小$w$；数据缓冲区$D^*=\varnothing$；概念漂移警告为$\delta$；串行基分类器上限$k$；交叉基分类器上限$f$；串行集成$H=\varnothing$；交叉集成$H^*=\varnothing$；概念漂移标记flag=false。

输出：串行集成$H=\{h_1,\cdots,h_k\}$；交叉集成$H^*=\{h_1^*,\cdots,h_f^*\}$；模型测试精度Acc。

(1) While 数据序列SD未结束

(2) 使用集成H_{t-2}与H_{t-1}在对应数据块D_{t-1}与D_t上预测，得到Acc_{t-1}与Acc_t；

(3) 根据式(3)更新H_{t-1}中基分类器的权重；

(4) if $\tilde{P}_{\mathrm{det}}^t<\delta$ and flag = false

(5) flag = true；

(6) $\mathrm{Acc}_A=\mathrm{Acc}_{t-1}$；

(7) end if

(8) if flag = true

(9) if $\tilde{P}_t<\delta$

创建数据缓冲区D^*,划分得到子数据块D_i^*,训练得到交叉基分类器h_i^*;
(10) else
(11) flag = false;
(12) end if
(13) end if
(14) 在D_t上训练得到h_t,根据式(4)初始化权重;
(15) if 串行分类器个数未达k
(16) 将h_t添加到H_{t-1}中;
(17) else
(18) 替换掉H_{t-1}中权重最小的基分类器;
(19) end if
(20) end while

3 实验与性能分析

为验证所提SC_ensemble算法的有效性,本文使用了具有不同类型的概念漂移数据集进行实验,分别从不同的评价指标对实验结果进行分析说明。

3.1 数据集

(1)合成数据集:为了检验算法对概念漂移的处理能力,本文使用大规模在线分析平台[28]中的流数据生成器MOA产生了具有概念漂移的数据集。旋转超平面数据集通过改变数据样本特征的权值来改变超平面的方向和位置,实验生成增量式概念漂移的数据集。LED数据集包含一个突变式漂移数据集LED_abrupt(漂移位点位于50 KB处)和一个渐进式漂移数据集(LED_gradual),漂移位点分别位于25 KB、50 KB和75 KB处。RBFlips数据集通过随机径向函数产生固定数目的随机中心,通过随机选择中心生成样本,该数据集概念漂移位点位于25 KB、50 KB和75 KB处。Sea是经典的突变式数据集,实验生成包含3次突变式概念漂移的数据集,漂移位点分别位于25 KB、50 KB和75 KB处。Tree数据集利用决策树生成数据,通过为每个子叶上的属性生成随机数的方式产生实例,该数据集的概念漂移位点位于25 KB、50 KB和75 KB处。

(2)真实数据集:实验中采用了网络入侵检测数据集KDDcup99、电力价格分析数据集Electricity、森林覆盖分析数据集Covertype以及天气数据集Weather。数据集的具体信息见表2,式中"—"表示不确定漂移位点数量或漂移位点位置。

表2 实验采用的数据集

Table 2 Datasets used in experiment

数据集	属性维数	样本类别数	样本数量	漂移类型	漂移位点数	位点位置
Hyperplane	10	2	100 KB	增量型	—	—
LED_abrupt	24	10	100 KB	突变型	1	50 KB
LED_gradual	24	10	100 KB	渐变型	3	25 KB,50 KB,75 KB
RBFblips	20	4	100 KB	突变型	3	25 KB,50 KB,75 KB
Sea	3	2	100 KB	渐变型	3	25 KB,50 KB,75 KB
Tree	30	10	100 KB	突变型	3	25 KB,50 KB,75 KB
KDDcup99	41	23	4.94 MB	Unknow	—	—
Electricity	6	2	45.3 KB	Unknow	—	—
Covertype	54	7	581 KB	Unknow	—	—
Weather	9	3	95.1 KB	Unknow	—	—

3.2 参数设置

为了充分验证所提出方法应对概念漂移的有效性，本文在不同参数下进行了实验研究。

(1) 概念漂移警告δ。为了能够准确检测出概念漂移位点，且较大的δ会将波动稍大的点错误检测为概念漂移位点，会发生严重的误检情况，较小的δ无法检测到波动较小的位点，会发生检测不到概念漂移位点的情况。因此，本文将δ的值设置为0.8。

(2) 数据块大小w。由于在过小数据块上无法得到足够多数据的样本特征，训练的分类器稳定性较差，而过大的数据块可能会在数据块中包含概念漂移，影响模型的分类效果。因此本文将w统一设置为100。

(3) 交叉分类器个数f。若f过小，则创建的交叉基分类器越少，从而能提取到的局部有效信息会减少，造成其交叉分类器的性能较低，这时交叉集成的性能会被控制全局性能的串行集成性能所抑制，导致没有发挥出交叉集成的优势；若f过大，则创建的交叉基分类器的个数越多，导致交叉集成中包含了过多的强分类器，从而抑制了集成性能。因此，交叉分类器的取值起着决定性作用，本文将其设置为$f=\{5, 10, 20\}$。

本文选择LIBSVM作为基分类器来构建“同质”基分类器。核参数采用默认值($g=1/m$，m为数据特征维度)，由于在复杂的非线性问题中，惩罚因子对模型性能的影响较大，因此，本文将惩罚因子设置为$C=\{1, 10, 100\}$。通过网格调参的方式，将不同的C与f相组合，比较不同参数组合下模型的分类性能，从而找出对应的全局最优参数组合，使用该组合与对比方法进行比较。

3.3 对比方法分析

为了更好地有效评估所提SC_ensemble方法的性能，本文选取了以下5种方法进行对比。

(1)SEA[15](A streaming ensemble algorithm (SEA) for large-scale classification)。传统流数据集成分类方法，是最早将集成学习应用到流数据分类中的方法，该方法使用固定大小的数据块构建基分类器并进行集成学习以适应概念漂移。

(2)AUE2[17](Reacting to different types of concept drift: the accuracy updated ensemble algorithm)。精度更新集成算法，利用模型精度的变化情况不断更新基分类器，通过不断增量更新历史基分类器来适应概念漂移。

(3)DWCDS[9](A double-window-based classification algorithm for concept drifting data streams)。该方法是一种基于双窗口机制的漂移检测方法，利用滑动窗口检测数据分布变化，动态更新学习模型。

(4)HBP[29](Hedge backpropagation)。反向传播算法是一种处理流数据的在线深度学习算法。它将深度神经网络的不同层次进行加权集成，并依据各个层次在各个时间步上的表现更新权值，以此来适应流数据分布的变化。

(5)ResNet[30](Deep residential network)。它是常见的深度学习算法，通过残差连接的方式构建深度残差单元，叠加多个深度残差单元的方式构建残差网络，有效地适应了流数据的变化。

3.4 评价指标

为了评估所提SC_ensemble算法的性能，本文针对模型的分类准确、模型对概念漂移的适应性能以及算法的稳定性，分别提出了以下4种不同的评测指标。

(1) AvgRAcc(Average real accuracy)。AvgRAcc表示模型在每个时间戳的实时精度的平均值，该指标用于反应模型的实时性能，是分类结果的有效衡量标准，其定义为

$$\mathrm{AvgRAcc}=\frac{1}{T}\sum_{t=1}^{T}\frac{\mathrm{TP}+\mathrm{TN}}{|D_t|} \tag{9}$$

式中：TP、TN分别表示正确分类的正类样本数和正确分类的负类样本数；$|D_t|$表示1个数据块中的样本

总数;T表示所有时间步的总和,越高的测试精度表明模型的分类性能越好。

(2) CumAcc(Cumulative accuracy)。CumAcc反应了模型在过去所有时间步上的性能,其定义为

$$\mathrm{CumAcc}=\frac{1}{T\times n}\sum_{t=1}^{T}n_t \tag{10}$$

式中:n_t表示时间步t时预测正确的样本数量;n表示每个时间步的样本数。

(3) RSA(Recovery speed under accuracy)。RSA表示数据分布发生变化后,模型的恢复能力,其定义为

$$\mathrm{RSA}=\mathrm{step}\times\mathrm{avge} \tag{11}$$

式中:step表示模型从概念漂移位点到收敛位点所用的步数;avge表示整体错误率的均值。RSA的值越小,表示在发生概念漂移后,考虑全局错误率较低的同时模型能够在较短的时间内做出快速恢复,表明其模型的收敛性越好。

(4) 鲁棒性[31]。鲁棒性是对模型稳定性能的评估,决定了学习模型的泛化性能,为了衡量不同算法在不同数据集的表现性能,本文对不同算法进行了鲁棒性分析(Robust-ness analysis),具体定义为

$$R_A(D)=\frac{\mathrm{Acc}_A(D)}{\min_\alpha \mathrm{Acc}_\alpha(D)} \tag{12}$$

式中:$\mathrm{Acc}_A(D)$表示算法A在数据集D上的分类准确率;$\min_\alpha \mathrm{Acc}_\alpha$表示在数据集$D$上所有算法的准确率的最小值。由式(12)可知,有最小精度的算法在数据集D上的$R_A(D)$为1,其余算法的$R_A(D)$值均大于1。假设有n个数据集$(D_1,\cdots,D_n)$,A算法在所有数据集上的鲁棒性定义为

$$R_A=\sum_{i=1}^{n}R_A(D_i) \tag{13}$$

由以上定义可知,R_A值越大,表明其算法的稳定性越好(即泛化性能越好)。

3.5 实验结果与分析

为验证本文所提出SC_ensemble算法的合理性以及该方法检测概念漂移与适应概念漂移的性能,本文从消融效果、模型测试精度、概念漂移适应情况以及鲁棒性方面分别进行测试分析。

3.5.1 消融效果分析

为了更好地验证本文在概念漂移后插入交叉基分类器的有效性,本节分别采用串行集成、交叉集成和串行交叉混合集成的方式对流数据的分类情况进行分析。表3展示了不同情况下模型的平均分类准确率,从表中可以看出只有交叉集成的模型性能较差,串行和交叉混合集成的模型性能表现最佳,其原因在于若在每一时刻都插入交叉分类器,则会使得集成模型中的子模型都具有较好的表现性能,而一批过强适应性的子模型反而会对模型整体的集成效果起到一定的抑制作用,因此,串行集成整体效果会好于交叉集成的整体效果。而对于串行集成来讲,虽然没有较多的过强子模型抑制整体集成性能,但是当数据分布发生较大变化时,不能及时提取出代表最新数据分布的局部有效信息,加入交叉集成可以在保持其整体有效信息的同时提取局部有效信息,可以加快模型对新数据分布的适应能力,以此来提高模型

表3 消融效果比较

Table 3 Comparison of ablation effects

数据集	串行集成	交叉集成	串行交叉混合集成
Hyperplane	0.913 2	0.867 6	0.913 3
LED_abrupt	0.588 4	0.465 5	0.589 6
LED_gradual	0.606 5	0.475 5	0.605 9
RBFblips	0.944 1	0.909 6	0.948 6
Sea	0.826 2	0.738 4	0.826 3
Tree	0.600 9	0.504 7	0.604 3
Kddcup99	0.938 3	0.846 6	0.938 3
Electricity	0.657 1	0.629 0	0.715 1
Covertype	0.649 1	0.489 7	0.658 0
Weather	0.878 1	0.866 1	0.892 5

的泛化性能。LED_gradual数据集上出现串行集成性能稍好的情况可能由于该数据集是渐变型的数据集,数据的波动小,造成概念漂移的漏检,导致没有提取到最新数据分布的有效信息,从而造成其模型的性能下降。在其余所有数据集上,实验结果均与预期相符。这也充分验证了本文将串行集成和交叉集成融合的有效性以及合理性。

3.5.2 模型精度分析

本节采用AvgRAcc评价指标,分别测试本文所提出的SC_ensemble算法在不同参数下的分类准确率情况,并与对比方法进行研究分析。

表4给出了不同参数下的结果,在大部分数据集上,随着C的增大,模型的平均测试精度随之增大,这是由于过小的C会导致欠拟合现象,并且为了防止C过大造成过拟合现象,本文将C的取值设定为100。其次,随着交叉分类器个数f的增大,模型精度出现下降趋势,这是由于f值越大,会有较多的强分类器抑制集成效果,导致过拟合发生。本文中取参数f=10,C=100与对比方法比较,对比方法的参数均采用默认值。

表4 不同参数下的平均准确率结果

Table 4 Results of AvgRAcc under different parameters

数据集	f=5			f=10			f=20		
	C=1	C=10	C=100	C=1	C=10	C=100	C=1	C=10	C=100
Hyperplane	0.899	0.914	0.913	0.899	0.914	0.913	0.899	0.914	0.913
LED_abrupt	0.479	0.591	0.591	0.475	0.589	0.590	0.472	0.586	0.588
LED_gradual	0.489	0.607	0.607	0.487	0.588	0.606	0.484	0.604	0.605
RBFblips	0.704	0.899	0.947	0.703	0.900	0.949	0.703	0.900	0.949
Sea	0.823	0.826	0.826	0.823	0.826	0.826	0.823	0.826	0.826
Tree	0.398	0.579	0.603	0.396	0.580	0.604	0.396	0.580	0.604
Kddcup99	0.934	0.938	0.938	0.934	0.938	0.938	0.934	0.938	0.938
Electricity	0.618	0.658	0.715	0.622	0.660	0.715	0.623	0.662	0.715
Covertype	0.627	0.647	0.668	0.616	0.638	0.658	0.611	0.63	0.648
Weather	0.883	0.883	0.884	0.892	0.892	0.893	0.892	0.892	0.892

由于数据量较大,实时精度较为密集,为了方便观察,本文通过CumAcc指标对不同方法的预测精度进行分析。

图6给出了不同方法在各个数据集上的CumAcc值。从实验结果可以看出,在合成数据集上,本文所提方法SC_ensemble算法在初始时间点具有较高的精度,在概念漂移发生后SC_ensemble下降趋势低于其他方法,这是由于该方法在平稳流数据环境下使用串行集成进行在线学习,模型具有较高的泛化性能,概念漂移发生后,在最新数据分布上创建了交叉基分类器,提高了模型的整体性能。从实验结果可以看出,在真实数据集上,SC_ensemble算法的模型性能稍逊于DWCDS,但是,相比于其他4个算法,本文算法的模型性能具有明显优势。

图7为不同方法在各个数据集上的平均准确率变化情况,从实验结果可以看出,SC_ensemble算法的模型整体性能在合成数据集上明显优于其他概念漂移检测及适应方法,这说明在包含突变式概念漂移和渐近式概念漂移的数据集上,本文提出的算法取得最高的准确率,表现出了良好的整体模型性能。在Covertype真实数据集上也表现出了良好的模型预测性能,虽然在其余3个真实数据集上本文所提方法的平均精度略低于DWCDS,这与数据集本身的特性有一定的关系,数据集Kddcup99、Electricity、Weather的分布极度倾斜,数据本身的波动变化较小,DWCDS方法采用双滑动窗口检测概念漂移,在

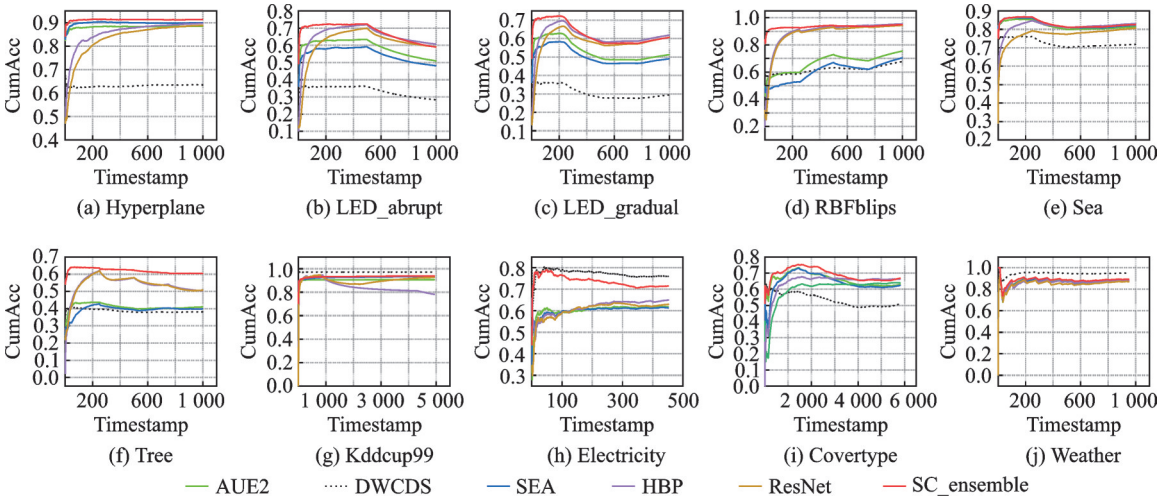

图6 不同方法的累计精度比较

Fig.6 Comparison of CumAcc of different methods

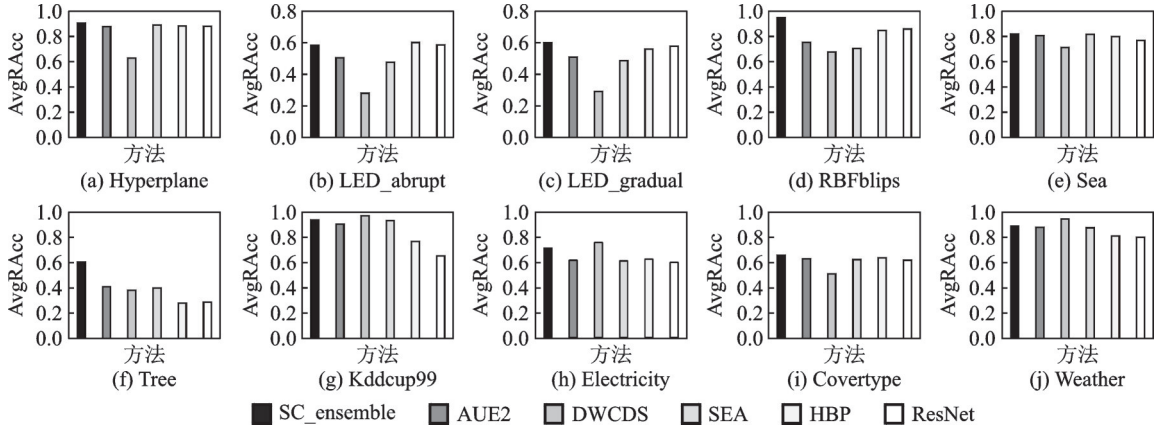

图7 不同方法的平均准确率比较

Fig.7 Comparison of AvgRAcc of different methods

数据波动较小时占有一定优势,而SC_ensemble算法在数据倾斜或较小波动的情况下,可能会发生概念漂移位点的漏检,会直接影响交叉分类器的插入,从而导致模型精度的下降。但是相比于其余4个方法而言,SC_ensemble算法仍表现出了较好的整体性能。

表5给出了不同方法在不同数据集上的平均精度排名情况,排名最高的数据均由粗体标出。从整体结果来看,SC_ensemble取得最高的准确率,HBP算法的整体排序与AUE2相当,SEA紧随其后,DWCDS与Resnet排名最后。从表中可以看出,在合成数据集上,除了RBFblips数据集上HBP算法取得最优值外,其余合成数据集上SC_ensemble算法均取得最高的准确率,这表明SC_ensemble算法能够有效处理各种类型的概念漂移,这是由于SC_ensemble算法在概念漂移后有效提取了最新数据分布的信息,充分发挥了交叉分类器的作用,使模型快速收敛,有效提高了模型的整体表现性能。在真实流数据环境下,Covertype数据集上,本文算法表现最好,在其余3个数据集上,DWCDS平均准确率最高,其次是SC_ensemble,这表明SC_ensemble算法能够较好地处理真实流数据,但是对于具有倾斜数据分布且数据本身波动较小的数据集的处理能力还有待提高。这主要是由于在波动较小的数据集上,比较容

表5 不同方法的平均准确率排序比较

Table 5 Ranking comparison of AvgRAcc on different methods

数据集	SC_ensemble	AUE2	DWCDS	SEA	HBP	ResNet
Hyperplane	0.913(1)	0.885(5)	0.635(6)	0.899(2)	0.890(3)	0.889(4)
LED_abrupt	0.590(3)	0.510(4)	0.283(6)	0.481(5)	0.607(1)	0.591(2)
LED_gradual	0.606(1)	0.513(4)	0.294(6)	0.491(5)	0.565(3)	0.584(2)
RBFblips	0.949(1)	0.754(4)	0.677(6)	0.705(5)	0.848(3)	0.859(2)
Sea	0.826(1)	0.814(3)	0.718(6)	0.823(2)	0.806(4)	0.776(5)
Tree	0.604(1)	0.410(2)	0.382(4)	0.399(3)	0.280(6)	0.288(5)
Kddcup99	0.938(2)	0.906(4)	0.971(1)	0.934(3)	0.767(5)	0.654(6)
Electricity	0.715(2)	0.618(4)	0.760(1)	0.613(5)	0.626(3)	0.601(6)
Covertype	0.658(1)	0.631(3)	0.511(6)	0.624(4)	0.639(2)	0.619(5)
Weather	0.893(2)	0.882(3)	0.948(1)	0.878(4)	0.814(5)	0.803(6)
平均排序	1.50	3.60	4.30	3.80	3.50	4.30

易发生概念漂移位点的漏检,影响交叉分类器的插入,从而影响算法性能。

本文使用非参数检验方法Friedman-Test[32],对所提算法与对比方法的性能优势进行统计检验分析。对于给定的k种方法和n个数据集,令r_i^j为第j个算法在第i个数据集上的秩,则第j个算法的秩和平均为$R_j=\frac{1}{n}\sum r_i^j$。零假设$H_0$假定所有方法性能是相同的。在此前提下,当$n$和$k$足够大时,Friedman统计值$F_F$服从第1自由度为$k-1$,第2自由度为$(k-1)(n-1)$的$F$分布,有

$$F_F=\frac{(n-1)\chi_F^2}{n(k-1)-\chi_F^2} \tag{14}$$

$$\chi_F^2=\frac{12n}{k(k+1)}\left[\sum_j R_j^2-\frac{k(k+1)^2}{4}\right] \tag{15}$$

若得到的统计值大于某一显著水平下F分布临界值,则拒绝零假设H_0,表明各算法的秩存在显著差异,反之接受零假设H_0,所有算法的性能无明显差异。对上述不同算法的平均准确率进行统计检验,可得Friedman在所有数据集上的统计值$F_F=3.995$,在$\alpha=0.05$的情况下F分布临界值为2.422,因此,拒绝零假设H_0,所有方法性能存在显著差异。

本文还通过Bonferroni-Dunn测试计算了所有方法的显著性差异,用于比较两个方法之间是否存在显著差异。若两种方法的秩和平均差值大于临界差,则这两种方法的性能存在显著差异,有

$$\mathrm{CD}=q_\alpha\sqrt{\frac{k(k+1)}{6n}} \tag{16}$$

式中:q_α为显著水平α下的临界值,经查表可得。通过计算可得,在所有数据集上,显著性水平$\alpha=0.05$的情况下CD=2.155 2。统计分析结果如图8所示。图中将没有显著性差异的方法使用黑线连接起来。统计分析结果表明,本文所提SC_ensemble方法在平均准确率上显著优于AUE2、SEA、DWCDS、HBP和ResNet方法。

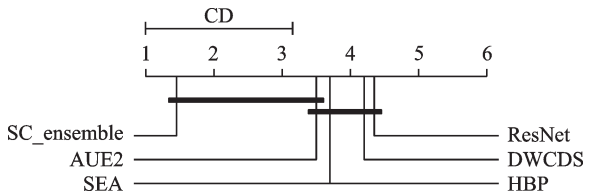

图8 不同方法平均准确率的显著性差异分析

Fig.8 Comparison of AvgRAcc against other methods with the Bonferroni-Dunn test

3.5.3 模型收敛性能分析

为了验证SC_ensemble算法在概念漂移发生后的收敛速度,本节比较并分析了不同概念漂移位点处的恢复性能。本文选取已知概念漂移位点的5个合成数据集进行分析。

表6给出了不同方法在不同位点处的恢复度,表中每个方法在每个对应数据集下的3个值分别表示前期、中期和后期概念漂移位点处的恢复值,若数据集在对应位置无概念漂移位点,则将该处记为“—”。考虑到不同模型对数据分布的表征能力不同,对于渐进式概念漂移的数据集,模型的性能波动会比较小。对于此类数据集,本文将发生概念漂移后到下一个概念漂移位点期间的实时精度均值作为模型性能是否恢复的参考值,否则对于模型性能波动较大的数据集将发生概念漂移前的精度作为其参考值。若某位点的精度大于参考值的80%,则将该位点视为收敛位点。从表中可以看出,SC_ensemble算法除了在Sea数据集的后期、Tree数据集的前期和中期上恢复性能略低于AUE2和DWCDS之外,在其他数据集上的恢复性能明显优于其他5种方法。这是由于当数据分布发生较大变化时,SC_ensemble算法及时有效提取了最新数据分布的有关信息,以此来进行策略更新,加快了模型的收敛速度。

表6 不同方法的恢复度比较

Table 6 Comparison of RSA of different methods %

数据集	SC_ensemble	SEA	AUE2	DWCDS	HBP	ResNet
LED_abrupt	—/1.23/—	—/4.68/—	—/1.47/—	—/1.44/—	—/38.11/—	—/36.30/—
LED_gradual	0.39/0.39/2.73	0.51/2.55/3.57	0.49/2.45/3.43	2.84/2.13/2.84	29.22/18.89/17.78	25.49/28.13/15.72
RBFblips	0.05/0.10/0.05	0.60/0.90/0.15	0.50/1.00/0.25	0.68/0.32/0.32	0.68/0.32/0.32	1.40/2.19/0.80
Sea	0.17/0.17/0.68	0.18/0.54/0.72	0.19/0.19/0.38	0.56/0.28/0.28	0.46/0.71/0.96	2.07/0.25/0.50
Tree	0.80/0.80/0.80	3.60/4.20/2.40	0.59/4.13/3.54	1.24/0.62/1.24	100.07/51.72/61.68	80.09/41.05/63.91

3.5.4 算法鲁棒性分析

为验证本文所提SC_ensemble算法的稳定性,本节从平均实时精度出发分析各算法的鲁棒性。图9给出了SC_ensemble算法与其他5个对比方法在不同数据集上的鲁棒性以及各算法的整体鲁棒性。图中,每个矩形的高度代表该算法在一个数据集上的鲁棒性值,矩形上面的数字代表对应算法的整体鲁棒性。从图9可以看出,本文所提算法在大部分数据集上的鲁棒性值明显优于其他算法。从算法整体鲁棒性来看,本文所提算法的整体鲁棒性值排名第一,AUE2次之,SEA、HBP和ResNet紧随其后,DWCDS的鲁棒性值最低,这是由于集成学习的算法在在一定程度上提高了算法的泛化性能。另外,SC_ensemble算法在平稳流数据环境下,利用串行集成模型进行全局信息的提取,概

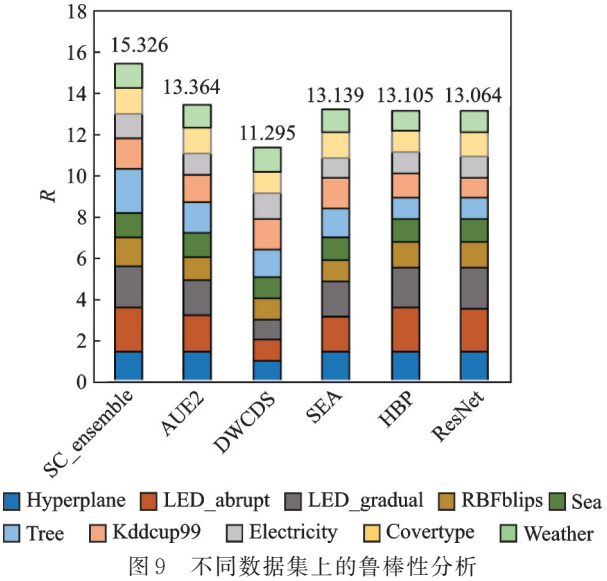

图9 不同数据集上的鲁棒性分析

Fig.9 Robustness analysis on different datasets

念漂移发生后通过串行集成与交叉集成的模型融合,在保持局部有效信息的同时兼顾全局有效信息,有效提高了算法的稳定性。

4 结束语

由于流数据中存在的概念漂移问题,常导致模型在漂移发生后对新分布数据不能快速适应,导致模型性能下降。本文提出了一种基于串行交叉混合集成的概念漂移检测及收敛方法,该方法借助集成学习思想,概念漂移发生后在漂移位点附近创建交叉分类器,兼顾了流数据包含的整体分布信息,又强化了概念漂移发生时的重要局部信息,使集成模型中包含了较多"好而不同"的基学习器,实现了漂移发生后学习模型的高效融合,使在线集成学习模型在概念漂移发生后能快速适应新分布的变化。在后续研究中,将针对不同类型的概念漂移问题,设计自适应的动态交叉分类器,即将交叉基分类器的个数设置为一个可自行调节的变量,使其随着数据波动大小的改变而动态变化。

参考文献:

[1] GEORG K, ZLIOBAITE I, BRZEZINSKI D. Open challenges for data stream mining research[J]. ACM SIGKDD Explorations Newsletter, 2014, 16(1): 1-10.

[2] LUGHOFER E, PRATAMA M. Online active learning in data stream regression using uncertainty sampling based on evolving generalized fuzzy models[J]. IEEE Transactions on Fuzzy Systems, 2018, 26(1): 292-309.

[3] 翟婷婷,高阳,朱俊武. 面向流数据分类的在线学习综述[J]. 软件学报,2020, 31(4): 912-931.
ZHAI Tingting, GAO Yang, ZHU Junwu. Survey of online learning algorithms for streaming data classification[J]. Journal of Software, 2020, 31(4): 912-931.

[4] 杜航原,王文剑,白亮. 一种基于优化模型的演化数据流聚类方法[J]. 中国科学:信息科学,2017, 47(11): 1464-1482.
DU Hangyuan, WANG Wenjian, BAI Liang. A novel evolving data stream clustering method based on optimization model[J]. Scientia Sinica: Information, 2017, 47(11): 1464-1482.

[5] MA J, SAUL L K, SAVAGE S, et al. Identifying suspicious URLs: An application of large-scale online learning[C]// Proceedings of the 26th Annual International Conference on Machine Learning. New York: ACM, 2009: 681-688.

[6] LU J, LIU A, DONG F, et al. Learning under concept drift: A review[J]. IEEE Transactions on Knowledge and Data Engineering, 2019, 31(12): 2346-2363.

[7] TENNANT M, STAHL F T, RANA O F, et al. Scalable real-time classification of data streams with concept drift[J]. Future Generation Computer Systems, 2017, 75: 187-199.

[8] 郭虎升, 张爱娟, 王文剑. 基于在线性能测试的概念漂移检测方法[J]. 软件学报, 2020, 31(4): 932-947.
GUO Husheng, ZHANG Aijuan, WANG Wenjian. Concept drift detection method based on online performance test[J]. Journal of Software, 2020, 31(4): 932-947.

[9] ZHU Q, HU X, ZHANG Y, et al. A double-window-based classification algorithm for concept drifting data streams[C]// Proceedings of the 2010 IEEE International Conference on Granular Computing. Piscataway, NJ: IEEE, 2010: 639-644.

[10] SUN Z, TANG J, QIAO J. Double window concept drift detection method based on sample distribution statistical test[C]// Proceedings of 2019 Chinese Automation Congress (CAC). Hangzhou: IEEE, 2019: 2085-2090.

[11] MARLON N, RAÚL F, RAFAEL M. Learning in environments with unknown dynamics: Towards more robust concept learners[J]. Journal of Machine Learning Research, 2007, 8(8): 2595-2628.

[12] DU L, SONG Q B, JIA X L. Detecting concept drift: An information entropy based method using an adaptive sliding window [J]. Intelligent Data Analysis, 2014, 18(3): 337-364.

[13] PESARANGHADER A, VIKTOR H L. Fast hoeffding drift detection method for evolving data streams[J]. Machine Learning and Knowledge Discovery in Databases, 2016, 9852: 96-111.

[14] BIFET A, GAVALDA R. Learning from time-changing data with adaptive windowing[C]//Proceedings of the 7th SIAM International Conference on Data Mining.[S.l.]: SDM, 2007: 443-448.

[15] STREET W N, KIM Y S. A streaming ensemble algorithm (SEA) for large-scale classification[C]//Proceedings of the 7th

ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, New York: ACM, 2001: 377-382.

[16] LU Y, CHEUNG Y M, TANG Y Y. Adaptive chunk-based dynamic weighted majority for imbalanced data streams with concept drift[J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(8): 2764-2778.

[17] BRZEZINSKI D, STEFANOWSKI J. Reacting to different types of concept drift: The accuracy updated ensemble algorithm [J]. IEEE Transactions on Neural Networks and Learning Systems, 2014, 25(1): 81-94.

[18] LIAO J W, DAI B R. An ensemble learning approach for concept drift[C]//Proceedings of 2014 International Conference on Information Science & Applications (ICISA). Seoul, Korea (South):[s.n.], 2014: 1-4.

[19] ELWELL R, POLIKAR R. Incremental learning of concept drift in nonstationary environments[J]. IEEE Transactions on Neural Networks, 2011, 22(10): 1517-1531.

[20] GUO H S, ZHANG S, WANG W J. Selective ensemble-based online adaptive deep neural networks for streaming data with concept drift[J]. Neural Networks, 2021, 142: 437-456.

[21] KOLTER J Z, MALOOF M A. Dynamic weighted majority: A new ensemble method for tracking concept drift[C]//Proceedings of the 3rd IEEE International Conference on Data Mining. Melbourne, FL, USA:[s.n.], 2003: 123-130.

[22] SIDHU P, BHATIA M, BINDAL A. A novel online ensemble approach for concept drift in data streams[C]//Proceedings of the 2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013).[S.l.]: IEEE, 2013: 550-555.

[23] BIFET A, HOLMES G, PFAHRINGER B, et al. New ensemble methods for evolving data streams[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.[S.l.]: ACM, 2009: 139-148.

[24] SHAN J, ZHANG H, LI W, et al. Online active learning ensemble framework for drifted data streams[J]. IEEE Transactions on Neural Networks and Learning Systems, 2019, 30(2): 486-498.

[25] OZA N C. Online bagging and boosting[C]//Proceedings of the IEEE International Conference on Systems, Man and Cybernetics.[S.l.]: IEEE, 2005: 2340-2345.

[26] OZA N C, RUSSELL S. Experimental comparisons of online and batch versions of bagging and boosting[C]//Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery Data Mining.[S.l.]: ACM, 2001: 359-364.

[27] BIFET A, HOLMES G, PFAHRINGER B. Leveraging bagging for evolving data streams[J]. Lecture Notes in Computer Science, 2010, 6231(1): 135-150.

[28] BIFET A, HOLMES G, KIRKBY R, et al. MOA: Massive online analysis[J]. Journal of Machine Learning Research, 2010, 11(52): 1601-1604.

[29] SAHOO D, PHAM Q, LU J, et al. Online deep learning: Learning deep neural networks on the fly[C]//Proceedings of the 27th International Joint Conference on Artificial Intelligence. [S.l.]: Polo Alto, 2017: 2660-2666.

[30] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.[S.l.]: IEEE, 2016: 770-778.

[31] 赵鹏, 周志华. 基于决策树模型重用的分布变化流数据学习[J]. 中国科学:信息科学, 2021, 51(1): 1-12.
ZHAO Peng, ZHOU Zhihua. Learning from distribution-changing data streams via decision tree model reuse[J]. Scientia Sinica: Information, 2021, 51(1): 1-12.

[32] DEMSAR J. Statistical comparisons of classifiers over multiple datasets[J]. Journal of Machine Learning Research, 2006, 7(1): 1-30.

作者简介:

郭虎升(1986-),男,博士,副教授,硕士生导师,研究方向:数据挖掘、机器学习和计算智能,E-mail:guohusheng@sxu.edu.cn。

高淑花(1996-),女,硕士研究生,研究方向:流数据挖掘和在线机器学习。

王文剑(1968-),**通信作者**,女,博士,教授,博士生导师,研究方向:机器学习、数据挖掘及计算智能,E-mail:wjwang@sxu.edu.cn。

(编辑:刘彦东)