计 算 机 研 究 与 发 展
Journal of Computer Research and Development

DOI: 10.7544/issn1000-1239.202220245
60(7): 1592−1602, 2023

基于在线集成的概念漂移自适应分类方法

郭虎升[1,2]　丛　璐[1]　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)

(guohusheng@sxu.edu.cn)

Adaptive Classification Method for Concept Drift Based on Online Ensemble

Guo Husheng[1,2], Cong Lu[1], Gao Shuhua[1], and Wang Wenjian[1,2]

[1](*School of Computer and Information Technology*, *Shanxi University*, *Taiyuan* 030006)

[2](*Key Laboratory of Computational Intelligence and Chinese Information Processing* (*Shanxi University*), *Ministry of Education*, *Taiyuan* 030006)

Abstract　In view of the problems that the online learning model cannot respond in time to the change of data distribution and it is difficult to extract the latest information of data distribution after concept drift occurs in streaming data, which leads to slow convergence of the learning model, an adaptive classification method for concept drift based on online ensemble (AC_OE) is presented. On the one hand, the online ensemble strategy is used to construct a local online learner, which can dynamically adjust the weight of base learner by local prediction of training samples in data blocks. It is helpful to not only extract the evolution information of streaming data in depth to make a more accurate response to the change of data distribution, but also improve the adaptability of the online learning model to the new data distribution after the occurrence of concept drift, and the real-time generalization performance of the learning model is improved too. On the other hand, the incremental learning strategy is used to construct a global incremental learner, and incremental training updates are carried out with the entry of new samples. The method extracts global distribution information of streaming data, and the model can maintain good robustness in the steady state of streaming data. Experimental results show that the proposed method can respond to concept drift and accelerate the convergence of online learning model, and improve the overall generalization performance of the learner effectively.

Key words　streaming data; concept drift; online ensemble; incremental learning; adaptive model

摘　要　针对流数据中概念漂移发生后,在线学习模型不能对分布变化后的数据做出及时响应且难以提取数据分布的最新信息,导致学习模型收敛较慢的问题,提出一种基于在线集成的概念漂移自适应分类方法(adaptive classification method for concept drift based on online ensemble, AC_OE).一方面,该方法利用在线集成策略构建在线集成学习器,对数据块中的训练样本进行局部预测以动态调整学习器权重,有助于深入提取漂移位点附近流数据的演化信息,对数据分布变化进行精准响应,提升在线学习模型对概念

收稿日期: 2022-03-24; **修回日期**: 2022-07-08

基金项目: 国家自然科学基金项目(62276157, U21A20513, 62076154, U1805263, 61503229);山西省自然科学基金项目(201901D111033);山西省重点研发计划项目(国际合作)(201903D421050)

This work was supported by the National Natural Science Foundation of China (62276157, U21A20513, 62076154, U1805263, 61503229), the Natural Science Foundation of Shanxi Province (201901D111033), and the Key Research and Development Program of Shanxi Province (International Cooperation) (201903D421050).

通信作者: 王文剑(wjwang@sxu.edu.cn)

漂移发生后新数据分布的适应能力，提高学习模型的实时泛化性能；另一方面，利用增量学习策略构建增量学习器，并随新样本的进入进行增量式的训练更新，提取流数据的全局分布信息，使模型在平稳的流数据状态下保持较好的鲁棒性. 实验结果表明，该方法能够对概念漂移做出及时响应并加速在线学习模型的收敛速度，同时有效提高学习器的整体泛化性能.

关键词 流数据；概念漂移；在线集成；增量学习；自适应模型

中图法分类号 TP18

大数据时代，流数据作为一种典型的数据类型受到广泛关注. 不同于传统静态数据，其具有动态性、时序性、无限性、不可再现性等特点，给数据的收集、存储、分析、处理，以及面向挖掘任务的模型构建和算法设计等都带来了严峻挑战[1-2]. 流数据在实际生产生活各领域的应用范围不断扩大，例如网络入侵、智慧医疗、气象预测等. 近年来，对流数据的研究备受关注，其目的是提高在线学习模型的泛化性能以适应流数据的实时分布[3-5]. 概念漂移是流数据挖掘在现实世界中的一个重要特性，也是流数据分析挖掘中不可避免的难点问题，它打破了传统机器学习中数据分布固定的假设，其典型特征是实时数据分布不断变化，并已受到越来越多的关注和研究[6-7].

流数据中存在的概念漂移使得由历史数据训练得到的学习模型很难适应分布变化后的新数据.例如在医学领域中，病毒可能会发生变异，若一直使用之前的在线学习模型对病毒进行筛查，很难在较短时间内发现变异后的病毒（如德尔塔毒株是由 COVID-19 病毒变异而来，随着病毒特征改变，需要实时更新筛查方法）；在气象预测领域，天气情况可能会受气温、空气湿度、压强等因素的影响，这些因素的改变都可能导致不同的天气情况，若无法检测气象因素的实时变化情况，就不能准确预测天气情况的变化. 因此，在含概念漂移的流数据挖掘中，需要打破传统机器学习对数据分布固定的假设，这对于提高在线学习模型的适应性能具有重要意义.

目前，利用集成学习处理概念漂移是非稳定环境下流数据挖掘采用的有效手段，即结合流数据的时序特性，构建多个具有差异性的基学习器，通过组合策略将多个弱学习器集成以形成一个性能较强的集成模型，提高学习模型的泛化性能. 然而，多数集成学习方法在漂移发生后不能对新数据分布做出及时响应，导致在线学习模型在漂移发生后不能快速收敛到新的分布，模型泛化性能较差.

为提高概念漂移发生后在线学习模型的快速响应能力及模型的实时泛化性能，本文提出一种基于在线集成的概念漂移自适应分类方法（adaptive classification method for concept drift based on online ensemble, AC_OE）. 该方法一方面通过在线集成策略对流数据进行局部的在线预测以实时调整基学习器权重，捕捉数据分布演化的局部细节信息；另一方面，引入增量学习器以获取流数据的全部分布信息，并随新样本的进入进行增量式的训练更新，提取流数据的全局分布信息，提高模型的鲁棒性. 所提出的方法使得概念漂移发生后在线学习模型能对概念漂移做出响应并快速收敛. 本文的主要贡献有 2 个方面：

1）通过对样本的在线局部预测，动态调整基学习器权重，提高模型对新分布数据的响应能力及收敛性能.

2）设置增量学习器以提取流数据的整体分布特征，提升在线学习模型的鲁棒性.

1 相关工作

目前，已有较多文献对流数据挖掘中概念漂移问题进行研究，常见的处理策略大致分为基于主动检测的概念漂移处理方法和基于被动自适应的概念漂移处理方法.

基于主动检测的概念漂移处理方法通过引入概念漂移检测机制，对流数据分布的稳定性进行检测或者通过模型实时性能指标（分类准确率、召回率等）的变化判断是否有概念漂移发生.当监测到数据分布不稳定或学习模型指标发生明显波动时，触发概念漂移警报，以及时对模型进行相应调整. 常见的主动检测方法主要有基于滑动窗口的方法和基于模型性能的方法.基于滑动窗口的方法采用单个或多个滑动窗口来存储处理数据，使用当前滑动窗口来容纳最新分布的样本，通过不断向前滑动窗口来判断是否有概念漂移的发生. 典型的方法有：使用自适应滑动窗口的熵方法[8]、基于自适应窗口的方法[9]、基于多窗口协同滑动的方法[10-11]. 基于模型性能的方法需要实时监测模型性能的变化情况，当检测到模型性能发生明显下降时，表明流数据中可能发生了概念漂移.典型的方法如：快速的概念漂移检测方法通过比较 2 次错误分类之间的标准差与设定阈值之间的大

小来检测概念漂移[12]；基于在线性能测试的方法通过比较漂移位点精度收敛偏差来判别概念漂移位点以及基于迁移学习的概念漂移检测方法[13-14].主动检测方法虽然能够在流数据非平稳状态下避免不必要的检测，提高了算法的效率，但是在学习过程中可能会发生概念漂移位点的误检、漏检及延检等情况，这将导致在线学习模型泛化性能降低.

基于被动自适应的概念漂移处理方法则不需要引入漂移检测机制来判断概念漂移的发生，而是通过不断调整学习器来适应数据分布的变化. 在被动自适应方法中，基于集成学习的处理方式较为常见，其根据学习单元大小可分为基于数据块的集成和基于单数据样本的在线集成.

基于数据块的集成每次对数据进行批量处理，典型的如：数据流集成方法将流数据分为固定大小的数据块，通过在数据块上训练基分类器构建集成模型，并根据一定的启发式规则使用最新数据块上构建的模型替换掉集成模型中性能最差的基分类器[15]；基于动态调整基分类器权重的方法通过不断调整基分类器的权重来适应概念漂移[16-18]；基于时序遗忘的方法使用遗忘机制对分类器进行动态加权[19]；基于选择性集成的在线自适应方法和基于迁移的集成学习方法[20-21]通过选择性集成及迁移学习技术提高基学习器的有效性.基于数据块集成的方法虽然能够很大程度上提高分类器的预测性能，但是当数据块中发生概念漂移时，模型不能做出快速响应，导致模型的收敛速度较慢.

基于单数据样本的在线集成方法每个时间戳仅处理1个样本进行模型更新，并对基分类器进行加权组合.典型方法如：基于混合标记策略的在线学习方法通过固定集成和动态集成相结合来适应概念漂移[22]；在线的Bagging方法和在线的Boosting方法将传统的集成学习技术改进应用于数据流的处理[23-24]. 基于单数据样本的在线集成方法虽然在一定程度上提高了模型对概念漂移的响应速度，但是难以提取重要的历史信息.

本文结合在线集成与增量学习策略，提出一种基于在线集成的概念漂移自适应分类方法.与传统方法相比，该方法既利用在线集成模型更新集成分类器权值，提升模型对局部演化特性的适应能力，又利用增量学习有效提取流数据的整体分布信息以提升鲁棒性，使学习模型在概念漂移发生后做出快速响应的同时提高收敛性能.

2 在线集成的概念漂移自适应分类方法

针对概念漂移发生后，在线学习模型不能做出及时响应且难以提取最新数据分布信息，导致模型收敛速度慢的问题，本文提出一种基于在线集成的概念漂移自适应分类方法.该方法有效结合在线集成与增量学习策略，通过在线集成对新到样本进行局部预测，更新在线集成模型中基学习器权重，以有效适应流数据的局部变化特性.同时结合增量学习对样本进行增量训练，提取流数据的整体分布信息，提升模型的鲁棒性. 图1为该方法的整体框架示意图.

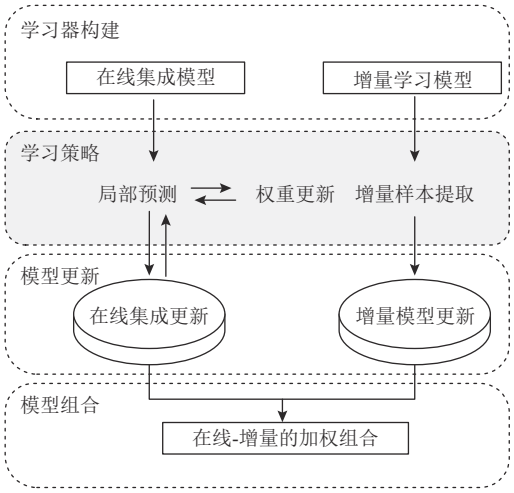

Fig. 1 The overall framework of AC_OE method

图1 AC_OE方法整体框架

2.1 问题定义

流数据是指随时间不断推移而产生的一系列具有实时性、持续性和不稳定性的数据，可以将其表示为：

$$S=\{(\boldsymbol{x}_i,y_i)\}_{i=1}^{t}, \tag{1}$$

其中$\boldsymbol{x}_i$是对应时刻的样本实例，y_i是该样本实例对应的标签. 若某一时刻流数据中样本的空间分布用一个“概念”来表示，则要学习的概念或者函数可用“目标概念”来表示. 假设流数据中数据的联合概念分布表示为$P(\boldsymbol{x},y)$，若在时刻t，流数据的概念分布发生变化，即该时刻发生了概念漂移，表示为：

$$\exists \boldsymbol{x}: P_{t-1}(\boldsymbol{x},y)\neq P_t(\boldsymbol{x},y). \tag{2}$$

2.2 在线集成的局部预测

由于流数据本身所具有的时序特性与集成学习机制高度契合，而集成学习又解决了单一学习器在流数据挖掘中不能把握全局信息的问题，因此通过在不同时刻构建基学习器进行集成，是流数据挖掘

的一条可行路径. 由于流数据中存在的概念漂移要求在线学习模型不仅能够对新数据分布快速收敛，也需要对概念漂移做出快速响应，而在线集成策略通过对样本进行逐个处理，有效提取漂移位点附近的细节分布演化特征，实现对概念漂移的快速响应. 因此，本文通过在线集成策略进行局部预测，并对基学习器权值进行更新，以使得在线集成模型适应概念漂移发生后流数据的快速变化，同时提高学习效率. 具体地，假设在线集成模型为 $H=\{(h_1,w_1),(h_2,w_2),\cdots,(h_k,w_k)\}$，其中，$h_i$为基学习器，$w_i$为所对应权值，初始状态下，每个基学习器对应权值$w_i=1/k$.

在流数据处于平稳状态时，通过反复训练后集成模型中的基学习器对新样本都保持较高的预测能力.然而，当概念漂移发生后，由于新样本分布发生变化，集成模型中的基学习器无法快速适应新的数据分布，对最新数据分布的预测能力较差. 因此，为对概念漂移快速响应，本文采用“权值在线更新、模型间隔训练”的方式，既通过在线的预测过程快速捕捉流数据当中数据分布的演化信息，并对集成学习的权值进行实时更新，同时又保持基学习器的相对稳定，通过每个数据块对基学习器进行替换，避免集成模型的不稳定波动影响学习性能. 具体地，假设数据流为$SD=\{D^1, D^2, \cdots, D^t, \cdots\}$，在时刻$t$当新样本$\boldsymbol{x}_j^t\in D^t$到达时，使用前序的集成模型$H$中的基学习器首先对其进行局部预测：

$$\tilde{y}_j^t=\sum_{i=1}^{k} w_i\cdot h_i(\boldsymbol{x}_j^t). \tag{3}$$

在线集成模型对时刻 t 的第 j 个样本的预测结果$\tilde{y}_j^t$与其真实标签y_j^t不相等时，则将该基学习器的权重根据式(4)作更新，并采用式(5)进行归一化；反之基学习器权重保持不变. 若流数据中发生概念漂移，在较短的时间内，在线集成中大多数历史基学习器的预测性能会保持较低的状态，则相应基学习器的权值会发生指数级下降，为了不使权重过低，本文在经过 1 个数据单元后对基学习器权重根据式(5)进行归一化处理，使其保持在区间 [0, 1] 内.

$$w_i=\beta\cdot w_i,\beta\in(0,1), \tag{4}$$

$$w_i=\frac{w_i}{\sum_{i=1}^{k} w_i}, \tag{5}$$

其中β表示权重衰退因子，若分类器将当前样本错误分类，则该分类器的权重将以一定步长减小. 当第t个时刻整个数据块D^t中样本全部处理完毕后进行基学习器的更新，即在D^t上训练得到新的学习器h，并选择在线集成模型中最差的基学习器进行替换：

$$h\leftrightarrow \arg\max_{h_i\in H}\sum_{j=1}^{n} h_i(\boldsymbol{x}_j^t)\neq y_j^t. \tag{6}$$

由于基学习器h是在最新的数据块D^t上训练得到的，代表着流数据的最新分布，因此将基学习器权值的初始值设置为 1. 在线集成的局部预测过程如图 2 所示，其中局部预测与权值更新是迭代进行的.

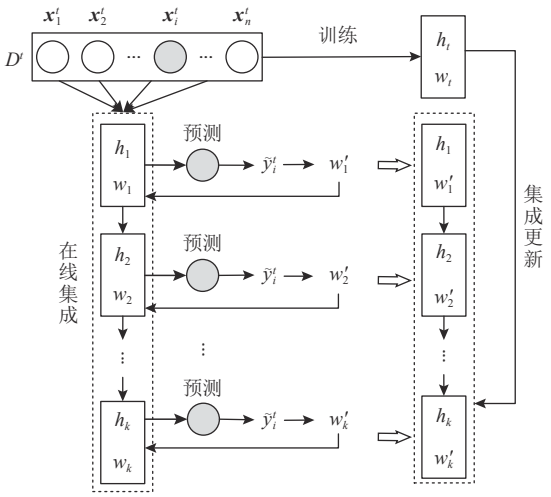

Fig. 2 Local prediction process for online ensemble

图 2 在线集成的局部预测过程

2.3 增量学习的全局预测

由于在线集成模型是在当前位点附近得到，其仅仅代表局部的分布信息.当流数据稳定时，仅采用在线集成模型无法覆盖整个流数据的分布信息，得到的基学习器鲁棒性较差. 因此本文同时构建增量学习器进行全局预测，以提取流数据的整体分布信息，提升鲁棒性. 具体地，在起始位点初始化 1 个增量学习器 I，在流数据进入过程中，根据每个新到达的数据块D^t以及历史数据块内的代表性关键样本，对增量学习器 I 进行增量更新，更新方式为：

$$I^t\leftarrow Tr(D^t\cup\{Rand(m,\sigma_{\tilde{y}_j^t=y_j^t}(\cup_{i=1}^{t-1}\{D^i\}))\}), \tag{7}$$

其中$Tr(\cdot)$表示在相应的数据集上训练得到学习器过程，$Rand(m,\cdot)$表示从样本中随机选择m个符合特定条件的样本(这里的特定条件$\sigma_{\tilde{y}_j^t=y_j^t}(\cdot)$指选择模型分类正确的样本). 增量学习的全局预测过程如图 3 所示.

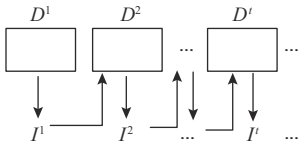

Fig. 3 Global prediction process for incremental learning

图 3 增量学习的全局预测过程

在此基础上，结合2.2节和本节所述的在线集成模型与增量学习模型构成总的测试模型，对待测样本进行加权投票：

$$y=\sum_{i=1}^{k} w_i\cdot h_i(\boldsymbol{x})+\alpha\cdot I(\boldsymbol{x}). \tag{8}$$

2.4 AC_OE方法

本文提出一种基于在线集成的概念漂移自适应分类方法，该方法通过在线集成来对新到样本进行局部预测，使模型对概念漂移及时响应，结合增量学习器做全局预测.利用历史数据块内适量关键样本与新到达的数据块内样本，分别提取关键历史信息与最新数据分布的信息，更新增量学习器，从而快速适应概念漂移. 在一个数据单元后，更新在线集成，提高了模型的整体泛化性能. 算法1展示了本文提出的AC_OE方法的执行流程. 对测试样本$\boldsymbol{x}$进行标签预测.

算法1. 在线集成的概念漂移自适应分类算法.

初始化：数据流$SD=\{D^1,\ D^2,\ \cdots,D^t,\ \cdots\}$，在线集成模型$H=\{(h_1,w_1),(h_2,w_2),\cdots,(h_k,w_k)\}$，$w_i=1/k$，增量学习模型$I$，权值衰退因子$\beta$.

① while 流数据SD未结束

第t个位点对应数据块D^t进入；

② while 对于D^t中新进入的第j个样本$\boldsymbol{x}_j^t$

③ 采用在线集成H中的每个基学习器h_i对当前样本$\boldsymbol{x}_j^t$进行局部预测，得到预测标签$h_i(\boldsymbol{x}_j^t)$；

④ if $h_i(\boldsymbol{x}_j^t)\neq y_j^t$

⑤ 更新在线集成的基学习器权重

$w_i=\beta\cdot w_i,\beta\in(0,1)$；

权值标准化 $w_i=\dfrac{w_i}{\sum_{i=1}^{k} w_i}$；

⑥ end if

⑦ end while

⑧ 在D^t上训练得到最新的在线基学习器h；

⑨ 选择H中最差的基学习器h^*

$h^*=\arg\max\limits_{h_i\in H}\sum\limits_{j=1}^{n} h_i(\boldsymbol{x}_j^t)\neq y_j^t$；

⑩ 用h替换h^*，得到最新的在线集成H；

⑪ 提取代表性关键样本并更新增量学习器

$I^t\leftarrow Tr(D^t\cup\{Rand(m,\sigma_{\tilde{y}_j^t=y_j^t}(\cup_{i=1}^{t-1}\{D^i\}))\})$；

⑫ 结合最新的在线集成H与增量集成I构建总的测试模型，并用于在测试集D^{t+1}上进行测试，得到实时精度；

⑬ end while

2.5 模型复杂度分析

本节将从时间复杂度与空间复杂度2个层面分析AC_OE方法的计算复杂度.

由于流数据挖掘的每一步过程主要的时间消耗在学习器的训练更新上，不妨假设传统在线学习模型在每个样本$\boldsymbol{x}$上进行训练更新学习器所需要的时间为$O(p^2)$，其中p表示每一步的模型训练更新所需要的样本规模，则传统在线学习模型的复杂度为$O(nTp^2)$，n为数据单元规模，T为流数据中总的数据块数. 而本文方法在在线集成中每个基学习器构建所需要的数据规模为n(一般地，$n<p$)，且每一个数据块仅需要训练更新1次在线基学习器，因此在线集成部分的复杂度为$O(Tn^2)$，尽管增量学习器构建所需要的时间复杂度与传统增量学习一致，但其与在线集成学习过程是并行的，因此所提出的模型时间复杂性相比于传统在线学习明显降低，同时又有效结合了在线集成与增量学习的优势.

在空间复杂度方面，AC_OE方法采取在线集成方式，每次仅利用最新数据块的样本更新基学习器，只增加1个数据块大小n的存储空间.此外，需要利用m个代表样本以及最新数据块内样本，实现增量学习器更新，因此需要增加m个存储空间，每次在线过程中需求的存储单元约为$O(n+m)$. 在线过程迭代执行时，不同时刻存储单元可以共用，因此，$O(n+m)$也为整个在线过程的空间复杂度. 然而，传统在线学习需要在整个数据流更新，因此，本文提出的方法空间复杂度低于传统在线学习.

3 实验与性能分析

为验证本文所提出的方法AC_OE对含概念漂移流数据的处理性能，本文在不同的含概念漂移的标准数据集和真实数据集上进行实验验证，实验平台为Windows10操作系统，CPU为酷睿i7-3. 2GHz内核，内存为8GB. 本方法采用MATLAB R2019a编写和运行. 与传统流数据集成分类算法(streaming ensemble algorithm, SEA)[15]、精度更新的集成算法(accuracy updated ensemble algorithm, AUE2)[17]以及深度神经网络(DNN-16)方法进行对比.

3.1 数据集

1)合成数据集.为检验算法处理概念漂移的能力，本文使用在线分析流数据生成器[25]来生成具有突变、渐变和增量类型的6个数据集.

① Hyperplane 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，d维空间的点$\boldsymbol{x}$的集合构成一个超平面：

$$\sum_{i=1}^{d} w_i \cdot x_i = w_0 = \sum_{i=1}^{d} w_i, \tag{9}$$

其中x_i是点$\boldsymbol{x}$的第i个坐标，$w_i \in [0, 1]$是相应权值. 当$\sum_{i=1}^{d} w_i x_i \geqslant w_0$时样本被标记为正类，否则被标记为负类.

② LED 数据集. 用来预测 7 段数码管上的数据，包含 24 个二进制属性；包含 1 个突变漂移数据集 LED_abrupt(漂移位点为 50×10^3)和 1 个渐变漂移的数据集 LED_gradual(漂移位点分别为 25×10^3, 50×10^3, 75×10^3).

③ RBFblips 数据集. 通过随机径向基函数生成固定数量的随机质心，每个质心包含其对应的随机位置、标准差、类别标签和权重；包含 3 个概念漂移位点，分别为 25×10^3, 50×10^3, 75×10^3.

④ Sea 数据集. 经典突变式漂移数据集，每个样本包含 f_1, f_2, f_3 共 3 个特征，其中类别只与前 2 个特征相关，当满足 $f_1+f_2<\theta$时，样本属于正类，反之属于负类；包含 3 个突变式概念漂移，位点为 25×10^3, 50×10^3, 75×10^3.

⑤ Tree 数据集. 通过决策树生成数据，为每个子叶上的属性生成随机数产生实例，概念漂移位点为 25×10^3, 50×10^3, 75×10^3.

2)真实数据集.除了合成数据集外，实验中还采用了 4 个真实数据集.

① KDDcup99 数据集[26]. 该数据集来自于第三届知识发现与数据挖掘竞赛，包括军事网络环境中模拟的各种攻击数据，用以检测网络入侵、区分正常的网络连接与恶意的网络连接.

② Electricity 数据集. 包含澳大利亚新南威尔州电力价格受天气、用户需求、供应情况和季节等因素影响的数据.

③ Covertype 数据集. 主要来自于美国林业局系统中某区域森林覆盖情况.

④ Weather 数据集. 覆盖了某地区 2006—2016 年的每日天气测量数据，包括温度、湿度、风向风速、能见度与大气压等，用于预测降雨情况.

实验中使用的数据集的详细信息如表 1 所示.

Table 1 Datasets Used in Our Experiment

表 1 本实验采用的数据集

数据集	属性维数	样本类别	样本数量	漂移类型	漂移位点数	漂移位点位置
Hyperplane	10	2	100×10^3	增量型	-	-
LED_abrupt	24	10	100×10^3	突变型	1	50×10^3
LED_gradual	24	10	100×10^3	渐变型	3	25×10^3, 50×10^3, 75×10^3
RBFblips	20	4	100×10^3	突变型	3	25×10^3, 50×10^3, 75×10^3
Sea	3	2	100×10^3	渐变型	3	25×10^3, 50×10^3, 75×10^3
Tree	30	10	100×10^3	突变型	3	25×10^3, 50×10^3, 75×10^3
KDDcup99	41	23	$4\,940\times10^3$	未知	-	-
Electricity	6	2	45.3×10^3	未知	-	-
Covertype	54	7	581×10^3	未知	-	-
Weather	9	3	95.1×10^3	未知	-	-

注: "-"表示不确定漂移位点数量或漂移位点位置.

3.2 参数设置

1)数据单元 n.为在合适时间间隔内对模型进行更新，较小的数据单元会导致挖掘效率较低；而较大的数据单元会导致模型更新不及时，对概念漂移的响应时速产生一定的影响，实验中数据单元 n =100.

2)衰退因子β.权值衰退因子β对于本文所提出的模型较为重要，衰退因子大时，容易减小漂移发生后不适用于新分布的基学习器，但容易导致模型发生振荡；反之，模型收敛较慢，无法快速适应漂移发生后新的数据分布，本文在不同的权值衰退因子β=0.8, 0.85, 0.9, 0.95 下进行了实验.

3)基学习器.实验中采用 LIBSVM 作为基学习器，核参数使用默认值(g =1/m, m 为数据特征维度)，在线集成中基学习器个数 k =10.

3.3 评价指标

为验证所提 AC_OE 方法的性能，本文从模型的准确率、模型的收敛性以及算法稳定性等方面进行了分析，具体指标有 4 个.

1)平均实时精度 *Avgracc*(average real-time accuracy). 表示模型的实时精度均值，反映模型整体分类性能，定义为：

$$Avgracc = \frac{1}{T}\sum_{t=1}^{T} racc_t, \tag{10}$$

其中 T 表示所有在线的时间步数，$racc_t$ 表示模型在时间戳 t 的实时精度. $racc_t$ 的计算公式为：

$$racc_t = \frac{n_t}{n}, \tag{11}$$

其中 n_t 表示时间戳 t 内能够正确分类的样本数，n 表示每个时间戳内的样本总数. $racc_t$ 越大表明模型实时性能越好.

2)累积精度 *Cumacc*(cumulative accuracy). 该指标反映了模型从开始到当前时刻的整体性能，定义为：

$$Cumacc = \frac{1}{T_t \times n}\sum_{t=1}^{T_t} n_t, \tag{12}$$

其中 T_t 表示当前累积的累积步数.

3)恢复值 *RSA*. *RSA* 衡量了在线学习模型在概念漂移发生后收敛到新分布数据所需的步数.

$$RSA = step \times (1 - avgracc), \tag{13}$$

其中 *step* 表示模型从概念漂移位点到收敛位点所需用的步数. 由于模型在不同概念漂移类型的数据集上的性能波动变化不同，本文采用漂移位点后的 n 个参考点的精度变化来判断该点是否为收敛点. 若 n 个参考位点的精度小于给定的阈值，且位点前 1/2 个和后 1/2 个参考位点的平均精度也小于阈值，则该位点被认为是收敛点. 收敛点的定义为：

$$\begin{aligned} &|racc_t - racc_{t+i}| < \varepsilon, \\ &\left|\frac{2}{n}\sum_{j=1}^{n/2} racc_{t+j} - \frac{2}{n}\sum_{k=(n/2)+1}^{n} racc_{t+k}\right| < \varepsilon, \end{aligned} \tag{14}$$

其中，$i \in \{1, 2, \cdots, n\}$.

4)鲁棒性 R(robustness)[27]. 鲁棒性是对模型稳定性能的有效评价指标，同时是模型泛化性能的体现，本文在平均精度上分析了不同算法的鲁棒性，算法 A 在不同数据集上的鲁棒性定义为：

$$R_A(D) = \frac{racc_A(D)}{\min\limits_{\alpha} racc_\alpha(D)}, \tag{15}$$

其中 $racc_A(D)$ 表示算法 A 在数据集 D 上的平均精度，$\min\limits_{\alpha} racc_\alpha(D)$ 表示在数据集 D 上所有算法中的最小平均精度值. 某算法的整体鲁棒性值为该算法在所有数据集上的鲁棒性值之和，假设有 n 个数据集，具体定义为：

$$R_A = \sum_{i=1}^{n} R_A(D_i). \tag{16}$$

3.4 实验结果与分析

为有效衡量所提出 AC_OE 方法的分类性能、发生概念漂移后模型的收敛效果和模型稳定性，本文从方法的平均实时精度、累积精度、概念漂移恢复性以及鲁棒性 4 个方面进行实验结果的分析.

3.4.1 模型实时精度分析

本节分析算法在不同衰退因子 β 和惩罚参数 C 下的表现性能. 表 2 展示了 AC_OE 方法在不同参数下的平均实时精度，可以看出，随着 β 的增大，平均实时精度大多出现先升后降趋势，这是由于 β 值直接影响基学习器权重的变化，过小的 β 会使权重下降速度过大，不能发挥出在线基学习器的性能，过大的 β 会使基学习器的权重下降速度变慢，在概念漂移发生后，对概念漂移的响应不及时. 随着 C 的增大，平均实时精度出现先上升后下降的趋势，这是由于过小的 C 会导致过拟合现象，过大的 C 会导致欠拟合. 本文方法在每个数据集上通过网格调参将不同的 C 与 β 组合，得出模型的最优参数组合，并在该组合下进行后续实验结果分析.

表 3 展示了不同方法在各个数据集上的平均实时精度以及综合排名情况. 从表 3 可以看出，在所有的真实数据集上，AC_OE 方法的模型性能最佳，在合成数据集上，除了在 Sea 数据集上 AC_OE 方法性能略低之外，在其他数据集上均有较好的排名；基于集成框架的算法能够提高模型的整体分类效果；在集成框架算法中，AUE2 和 SEA 都是基于数据块的集成，而 AC_OE 方法与使用在线集成的方法，在概念漂移发生时可以及时对概念漂移做出响应，从而提高模型的整体性能.

本文使用非参数检验方法 Friedman-Test[28]，对所提方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 个方法在第 i 个数据集上的秩，则第 j 个方法的秩和平均值为 $R_j = \frac{1}{n}\sum\limits_i r_i^j$. 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 服从第一自由度为 k–1，第二自由度为 $(k-1)\times(n-1)$ 的 F 分布：

$$F_F = \frac{(n-1)\chi_F^2}{n(k-1) - \chi_F^2}, \tag{17}$$

$$\chi_F^2 = \frac{12n}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right]. \tag{18}$$

当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

Table 2 Results of Average Real-time Accuracy Under Different Parameters

表 2 不同参数下平均实时精度

数据集	C=1				C=10				C=100			
	β=0.8	β=0.85	β=0.9	β=0.95	β=0.8	β=0.85	β=0.9	β=0.95	β=0.8	β=0.85	β=0.9	β=0.95
Hyperplane	0.867 1	0.865 4	0.866 2	0.868 7	0.886 9	0.889 7	0.894 5	**0.896 6**	0.870 3	0.876 1	0.884 2	0.889 3
LED_abrupt	0.392 9	0.394 2	0.395 3	0.399 1	0.477 1	0.479 6	0.486 6	**0.505 4**	0.468 1	0.470 0	0.477 2	0.499 9
LED_gradual	0.401 1	0.403 1	0.402 8	0.408 9	0.487 6	0.490 8	0.499 1	**0.517 8**	0.478 2	0.481 7	0.490 1	0.510 4
RBFblips	0.698 7	0.696 9	0.704 8	0.684 4	0.873 2	0.875 9	0.878 2	0.880 2	0.924 1	0.927 1	0.927 5	**0.931 6**
Sea	0.791 8	0.794 6	0.799 0	**0.802 7**	0.741 7	0.748 7	0.757 4	0.776 4	0.738 4	0.736 9	0.747 4	0.772 0
Tree	0.380 3	0.378 6	0.369 5	0.367 4	0.529 4	0.533 3	0.537 2	**0.548 0**	0.506 2	0.509 3	0.515 7	0.538 3
KDDcup99	0.943 2	**0.944 6**	0.939 5	0.880 5	0.942 6	0.910 6	0.936 9	0.890 2	0.924 6	0.910 6	0.936 9	0.890 2
Electricity	0.742 1	0.745 0	0.739 4	0.709 4	0.769 2	0.762 4	0.758 9	0.731 7	**0.791 9**	0.791 5	0.790 4	0.760 4
Covertype	0.638 7	0.640 9	0.739 5	0.722 9	0.649 5	0.761 0	0.766 0	0.760 8	0.764 7	0.764 7	**0.781 3**	0.779 6
Weather	0.906 3	0.901 0	0.897 9	0.903 6	0.905 1	0.902 8	0.898 2	**0.906 9**	0.906 0	0.902 9	0.898 3	**0.906 9**

注：C 表示惩罚参数，β表示衰退因子，黑体数字表示最高平均实时精度.

Table 3 Comparison of Average Real-Time Accuracy Under Different Methods

表 3 不同方法的平均实时精度比较

数据集	平均实时精度（排名）			
	AC_OE	AUE2	DNN-16	SEA
Hyperplane	0.896 6（2）	0.884 7（3）	0.882 1（4）	**0.899 2**（1）
LED_abrupt	0.505 4（2）	**0.509 6**（1）	0.376 1（4）	0.481 3（3）
LED_gradual	**0.517 8**（1）	0.512 7（2）	0.391 8（4）	0.490 5（3）
RBFblips	**0.931 6**（1）	0.751 9（3）	0.802 3（2）	0.704 5（4）
Sea	0.802 7（3）	0.813 9（2）	0.701 9（4）	**0.822 9**（1）
Tree	**0.548 0**（1）	0.410 1（2）	0.163 4（4）	0.398 6（3）
KDDcup99	**0.944 6**（1）	0.902 3（3）	0.301 7（4）	0.933 8（2）
Electricity	**0.791 9**（1）	0.618 3（2）	0.512 8（4）	0.613 2（3）
Covertype	**0.781 3**（1）	0.630 6（2）	0.626 9（3）	0.624 0（4）
Weather	**0.906 9**（1）	0.882 4（2）	0.804 3（4）	0.878 1（3）
平均排序	1.4（1）	2.2（2）	3.7（4）	2.7（3）

注：黑体数字表示最高平均实时精度 .

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 统计值在所有数据集上的统计值$F_F = 11.2703$，在显著水平$\alpha = 0.05$的情况下 F 分布临界值为 2.960，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 测试计算所有方法的显著性差异，用于比较 2 种方法之间是否存在显著差异. 若 2 种方法的秩和平均差值大于临界差 CD，则这 2 种方法的性能存在显著差异.

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6n}}, \tag{19}$$

其中q_α为显著水平α下的临界值，经计算可得，在所有数据集上，显著性水平$\alpha = 0.05$的情况下$CD = 1.382\,2$. 统计分析结果如图 4 所示，结果表明，在统计意义上，本文所提 AC_OE 方法具有明显的优势.

图 5 为各个数据集上的累积精度，可以看出 AC_OE 方法除了在 Sea 数据集上的累积精度略低于 SEA 和

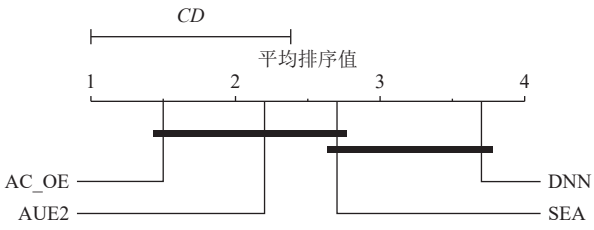

Fig. 4 Bonferroni-Dunn test result for average of different methods

图 4 不同方法平均精度的 Bonferroni-Dunn 检验结果

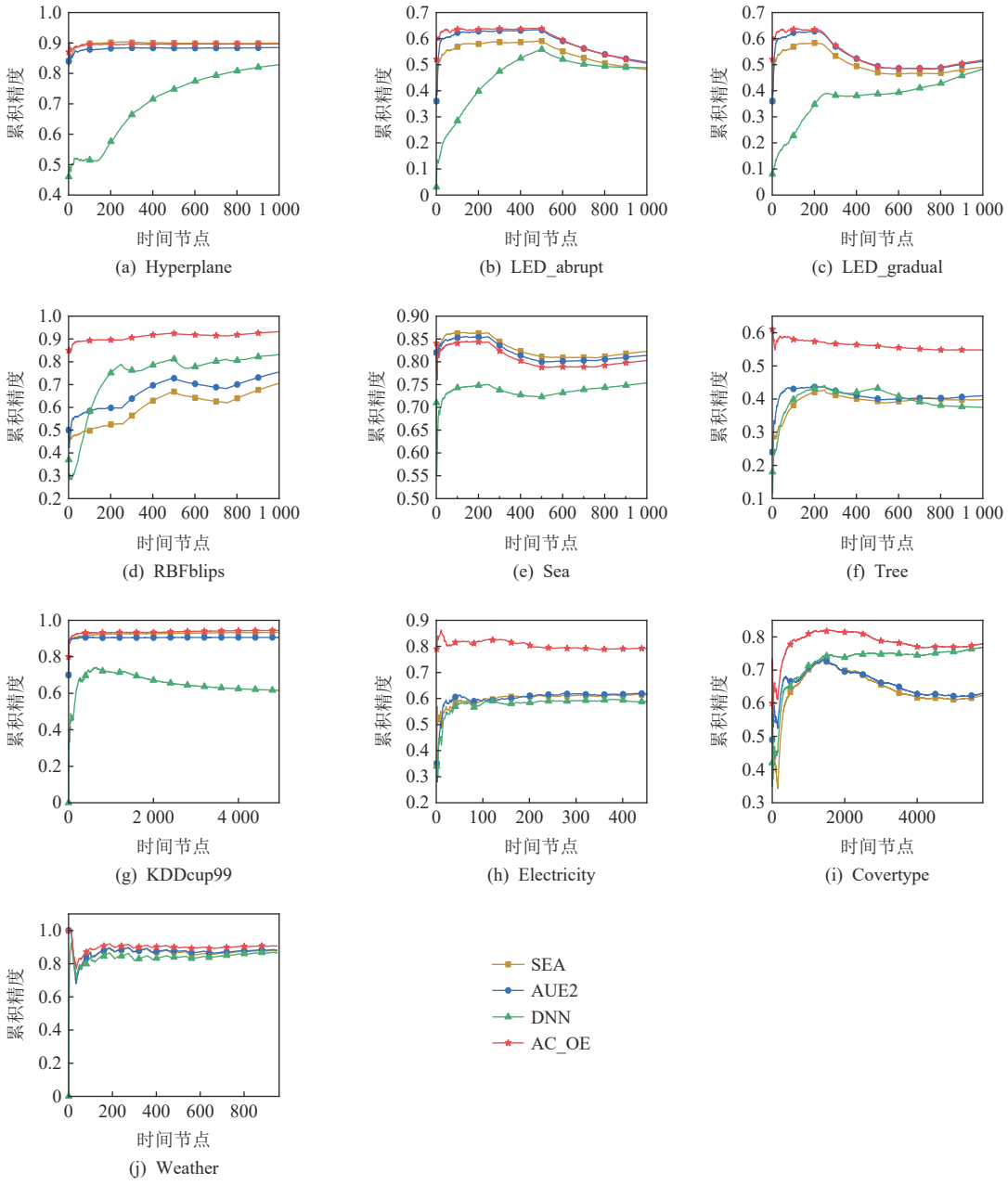

Fig. 5 Comparison of cumulative accuracy of different methods on every dataset

图 5 不同方法在各数据集上的累积精度比较

AUE2 算法，在其他数据集上均明显高于其他方法. 这是由于在概念漂移发生时，AC_OE 做出及时响应，保持较高的精度，使模型整体的性能得到了提高.另外，大多数数据集上，其他 2 个集成框架算法的整体性能相对较高，这说明集成学习具有较好的处理概念漂移的能力.

3.4.2 模型收敛性分析

当流数据发生概念漂移后，在线学习模型能否快速收敛到新的数据分布是衡量算法的重要指标. 表 4 展示不同方法在已知概念漂移位点的 5 个数据集上的模型收敛结果. 表 4 中每个算法对应的数据集的 3 个值分别代表前、中、后 3 个位点的恢复值；由于 Tree 数据集在中期位点精度下降后一直没有恢复，保持平稳波动，因此对中后 2 个位点的恢复度用“*”表示. 可以看出，除数据集 Sea 之外，本文所提出的 AC_OE 方法在其他 4 个数据集上的恢复值明显优于其他方法. 这是由于 AC_OE 方法能够充分利用最新样本，得到了代表最新数据分布的增量学习器，通过不断进行增量更新，能够使模型在漂移发生后快速收敛到新的数据分布，从而提升了模型的收敛性.

3.4.3 模型鲁棒性分析

鲁棒性是衡量算法稳定性的重要指标，值越大

Table 4 Recover value（*RSA*）Comparison of Different Methods

表 4 不同方法的恢复值（*RSA*）比较

数据集	AC_OE			SEA			AUE2			DNN		
	前位点	中位点	后位点	前位点	中位点	后位点	前位点	中位点	后位点	前位点	中位点	后位点
LED_abrupt	-	**1.02**	-	-	4.68	-	-	1.47	-	-	41.6	-
LED_gradual	**0.48**	**0.48**	**0.48**	0.51	2.55	3.57	0.49	2.45	3.43	74.0	99.5	40.6
RBFblips	**0.07**	**0.07**	**0.07**	0.60	0.90	0.15	0.50	1.00	0.25	5.20	22.8	3.48
Sea	0.20	0.20	0.40	**0.18**	0.54	0.72	0.19	**0.19**	**0.38**	1.02	3.90	3.90
Tree	**0.45**	**0.45**	**0.90**	3.60	4.20	2.40	0.59	4.13	3.54	96.5	*	*

注：“-”表示该位点处没有发生概念漂移，黑体数字表示最高平均实时精度，“*”表示不对当前位点恢复度进行统计.

表示模型越稳定. 图 6 展示了不同算法在不同数据集上的鲁棒性，不同的小矩形高度代表的是算法在不同数据集上的鲁棒性值的大小，每一列上面的数值表示该算法在所有数据集上的鲁棒性值的总和，即该算法的整体鲁棒性. 可以看出，在大多数数据集上，AC_OE 方法的鲁棒性均优于其他 3 种方法，且整体鲁棒性得到了最优值，AUE2, SEA 两个使用集成学习的方法也取得较好的结果，这是由于 AUE2, SEA 使用了集成学习框架，将多个弱分类器组合，提高了模型的整体泛化性能.

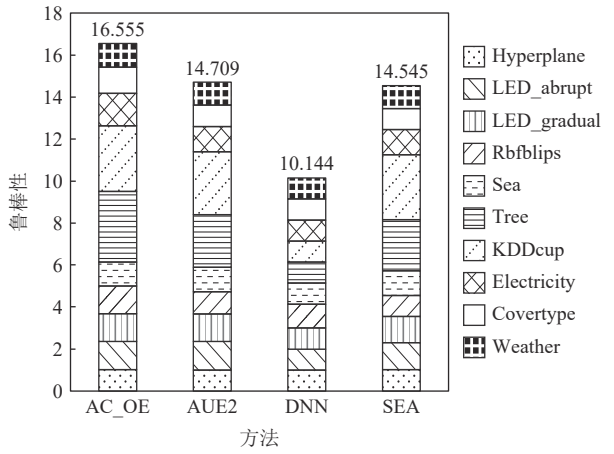

Fig. 6 Robustness comparison of different methods

图 6 不同方法的鲁棒性比较

4 结束语

针对流数据中概念漂移发生后，在线学习模型不能对分布变化后的数据做出及时响应且难以提取数据分布的最新信息，导致学习模型收敛较慢的问题，本文提出一种基于在线集成的概念漂移自适应分类方法. 一方面，该方法利用在线集成策略构建局部的在线学习器，对数据块中的训练样本进行局部预测以动态调整学习器权重，有助于深入提取漂移位点附近流数据的演化信息，以对数据分布变化做出更精准的响应，提升在线学习模型对概念漂移发生后新数据分布的适应能力，提高学习模型的实时泛化性能；另一方面，利用增量学习策略构建全局的增量学习器，并随新样本的进入进行增量式的训练更新，提取流数据的全局分布信息，使模型在平稳的流数据状态下保持较好的鲁棒性.

作者贡献声明：郭虎升负责思想提出、方法设计、初稿写作及论文修改；丛璐负责初稿写作、数据测试及论文修改；高淑花负责代码实现、数据测试及初稿写作；王文剑负责思想提出、写作指导、修改审定.

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] Zhai Tingting, Gao Yang, Zhu Junwu. Survey of online learning algorithms for streaming data classification[J]. Journal of Software, 2020, 31(4): 912−931 (in Chinese)
(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)

[4] Du Hangyuan, Wang Wenjian, Bai Liang. A novel evolving data stream clustering method based on optimization model[J]. SCIENTIA SINICA:Informationis, 2017, 47(11): 1464−1482 (in Chinese)
(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)

[5] Ma J, Saul L K, Savage S, et al. Identifying suspicious URLs: An application of large-scale online learning [C] // Proc of the 26th Annual Int Conf on Machine Learning, New York: ACM, 2009: 681−688

[6] Lu Jie, Liu Anjin, Dong Fan, 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] Du Lei, Song Qinbao, Jia Xiaolin. Detecting concept drift: An information entropy based method using an adaptive sliding window[J]. Intelligent Data Analysis, 2014, 18(3): 337−364
[9] Bifet A, Gavalda R. Learning from time-changing data with adaptive windowing [C] // Proc of the 7th SIAM Int Conf on Data Mining. Philadelphia, PA: SIAM, 2007: 443−448
[10] Guo Husheng, Li Hai, Ren Qiaoyan, et al. Concept drift type identification based on multi-sliding windows[J]. Information Sciences, 2022, 585: 1−23
[11] Guo Husheng, Ren Qiaoyan, Wang Wenjian. Concept drift class detection based on time window[J]. Journal of Computer Research and Development, 2022, 59(1): 127−143 (in Chinese)
(郭虎升, 任巧燕, 王文剑. 基于时序窗口的概念漂移类别检测[J]. 计算机研究与发展, 2022, 59(1): 127−143)
[12] Baena-García M, Campo-Ávila R J, Fidalgo D, et al. Early drift detection method [C] // Proc of the 17th ECML PKDD Int Workshop on Knowledge Discovery From Data Streams. Berlin: Springer, 2006: 77−86
[13] Guo Husheng, Zhang Aijuan, Wang Wenjian. Concept drift detection method based on online performance test[J]. Journal of Software, 2020, 31(4): 932−947 (in Chinese)
(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)
[14] Wen Yimin, Tang Shiqi, Feng Chao, et al. Online transfer learning for mining recurring concept in data stream classification[J]. Journal of Research and Development, 2016, 53(8): 1781−1791 (in Chinese)
(文益民, 唐诗淇, 冯超, 等. 基于在线迁移学习的重现概念漂移数据流分类[J]. 计算机研究与发展, 2016, 53(8): 1781−1791)
[15] Street W N, Kim Y S. A streaming ensemble algorithm (SEA) for large-scale classification [C] // Proc of the 7th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2001: 377−382
[16] Lu Yang, Cheung Y M, Tang Yuanyan. 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] Kolter J, Maloof M. Dynamic weighted majority: A new ensemble method for tracking concept drift [C] // Proc of the 3rd IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2003: 123−130
[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 Husheng, Zhang Shuai, Wang Wenjian. Selective ensemble-based online adaptive deep neural networks for streaming data with concept drift[J]. Neural Networks, 2021, 142: 437−456
[21] Sun Yu, Tang Ke, Zhu Zexuan, et al. Concept drift adaptation by exploiting historical knowledge[J]. IEEE Transactions on Neural Networks and Learning Systems, 2018, 29(10): 4822−1832
[22] Shan Jicheng, Zhang Hang, Li Weike, 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
[23] Oza N C. Online bagging and boosting [C] // Proc of the IEEE Int Conf on Systems, Man and Cybernetics. Piscataway, NJ: IEEE, 2005: 2340−2345
[24] Oza N C, Russell S. Experimental comparisons of online and batch versions of bagging and boosting [C] //Proc of the 7th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2001: 359−364
[25] Bifet A, Holmes G, Kirkby R, et al. MOA: Massive online analysis[J]. Journal of Machine Learning Research, 2010, 11(52): 1601−1604
[26] Sigkdd. KDDCup99 data [DB/OL]. [2019-04-19]. http://kdd.ics.uci.edu/data-baseskddcup99/kddcup99.htlm
[27] Zhao Peng, Zhou Zhihua. Learning from distribution-changing data streams via decision tree model reuse[J]. SCIENTIA SINICA:Informationis, 2021, 51(1): 1−12 (in Chinese)
(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)
[28] Demsar J. Statistical comparisons of classifiers over multiple datasets[J]. Journal of Machine Learning Research, 2006, 7(1): 1−30

Guo Husheng, born in 1986. PhD, associate professor, master supervisor. Senior member of CCF. His main research interests include data mining, machine learning, and computational intelligence.
郭虎升, 1986 年生. 博士, 副教授, 硕士生导师. CCF 高级会员. 主要研究方向为数据挖掘、机器学习和计算智能.

Cong Lu, born in 1998. Master candidate. Her main research interests include stream data mining and online machine learning.
丛　璐, 1998 年生. 硕士研究生. 主要研究方向为流数据挖掘和在线机器学习.

Gao Shuhua, born in 1996. Master. Her main research interests include stream data mining and online machine learning.
高淑花, 1996 年生. 硕士. 主要研究方向为流数据挖掘和在线机器学习.

Wang Wenjian, born in 1968. PhD, professor, PhD supervisor. Distinguished member of CCF. Her main research interests include machine learning, data mining, and computational intelligence.
王文剑, 1968 年生. 博士, 教授, 博士生导师. CCF 杰出会员. 主要研究方向为机器学习、数据挖掘及计算智能.